

**ĐẠI HỌC QUỐC GIA TP HCM  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA HỆ THÔNG THÔNG TIN**



**ĐỒ ÁN**

**Môn học: KHO DỮ LIỆU VÀ OLAP  
Đề tài: Xây dựng kho dữ liệu US ACCIDENTS**

**Giảng viên:** ThS. Nguyễn Thị Kim Phụng

**Lớp:** Kho dữ liệu và OLAP - IS403.O22.HTCL

**Thành viên:** Nguyễn Trương Đình Giang - 21520215

Nguyễn Thế Vinh - 21522794

TP Hồ Chí Minh, tháng 07 năm 2024

## MỤC LỤC

LỜI CẢM ƠN .....	5
NHẬN XÉT CỦA GIÁO VIÊN .....	6
CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ DỮ LIỆU .....	7
1. PHÁT BIỂU VỀ DỮ LIỆU.....	7
1.1. Mô tả về dữ liệu.....	7
1.2. Thuộc tính của kho dữ liệu .....	7
1.3. Kho dữ liệu đã xử lý .....	7
2. XÂY DỰNG KHO DỮ LIỆU.....	11
2.1. Sơ đồ hình sao minh họa .....	11
2.2. DIM_TIME.....	11
2.3. DIM_LOCATION.....	12
2.4. DIM_TRAFFIC_CONDITION.....	12
2.5. DIM_AIRPORT .....	14
2.6. DIM_WIND .....	14
2.7. DIM_TEMPERATURE .....	14
2.8. DIM_VISIBILITY .....	15
2.9. DIM_PRECIPITATION.....	15
2.10. DIM_WEATHER .....	15
2.11. FACT .....	15
CHƯƠNG 2. TÍCH HỢP DỮ LIỆU VÀ KHO (SSIS).....	16
1. CHUẨN BỊ CÔNG CỤ VÀ DATA WAREHOUSE.....	16
2. TẠO PROJECT SSIS TRONG VISUAL STUDIO 2022 .....	19
3. TẠO BẢNG DIM VÀ BẢNG FACT .....	20
3.1. Bảng DIM_TIME .....	21
3.2. Bảng DIM_LOCATION .....	25
3.3. Bảng DIM_TRAFFIC_CONDITION .....	27
3.4. Bảng DIM_AIRPORT .....	29
3.5. Bảng DIM_WIND .....	31
3.6. Bảng DIM_TEMPERATURE .....	33
3.7. Bảng DIM_VISIBILITY .....	35
3.8. Bảng DIM_PRECIPITATION .....	37
3.9. Bảng DIM_WEATHER .....	39
3.10. Bảng FACT .....	41
3.10.1. Merge Fact_Raw và Dim_Location vào Fact1 .....	42
3.10.2. Merge Fact1 và Dim_Traffic_Condition vào Fact2.....	47
3.10.3. Merge Fact2 và Dim_Airport vào Fact3 .....	51
3.10.4. Merge Fact3 và Dim_Wind vào Fact4 .....	55
3.10.5. Merge Fact4 và Dim_Weather vào Fact5 .....	59
3.10.6. Merge Fact5 và Dim_Visibility vào Fact6.....	63
3.10.7. Merge Fact6 và Dim_Temperature vào Fact7 .....	67
3.10.8. Merge Fact7 và Dim_Precipitation vào Fact .....	71
3.10.9. Tạo khóa ngoại từ bảng Fact đến các Dimension .....	76
4. CHẠY SSIS .....	76
CHƯƠNG 3. PHÂN TÍCH DỮ LIỆU TRONG KHO (SSAS) .....	80
1. XÁC ĐỊNH KHUNG NHÌN DỮ LIỆU NGUỒN (DEFINE DATE SOURCE VIEW).....	80
2. XÂY DỰNG CÁC KHỐI (CUBES) VÀ XÁC ĐỊNH CÁC ĐỘ ĐO (MEASURE)	
	85

## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

3.	XÁC ĐỊNH CÁC CHIỀU (DEFINE A DIMENSION).....	91
4.	XÁC ĐỊNH CÁC ĐỘ ĐO (MEASURES).....	97
5.	PHÂN CẤP TRONG BẢNG CHIỀU.....	99
5.1.	PHÂN CẤP TRONG BẢNG Dim_Time .....	99
5.2.	PHÂN CẤP TRONG BẢNG Dim_Location.....	112
5.3.	CHẠY DỰ ÁN SSAS .....	119
6.	THỰC HIỆN 10 CÂU TRU VÂN (MDX) .....	120
6.1.	Hiển thị danh sách các tiểu bang có dữ liệu tai nạn theo từng năm.....	120
6.1.1.	Thực hiện trên các khối Cubes. ....	120
6.1.2.	Thực hiện trên SQL. ....	121
6.1.3.	Thực hiện trên Excel. ....	122
6.1.4.	Thực hiện trên PowerBI. ....	125
6.2.	Top 5 tiểu bang có số vụ tai nạn lớn nhất của từng năm, sắp xếp giảm dần (TOP COUNT). ....	128
6.2.1.	Thực hiện trên các khối Cubes. ....	128
6.2.2.	Thực hiện trên SQL. ....	129
6.2.3.	Thực hiện trên Excel. ....	129
6.2.4.	Thực hiện trên PowerBI. ....	131
6.3.	Liệt kê các loại thời tiết có số vụ tai nạn từ 1000 trở xuống.....	133
6.3.1.	Thực hiện trên các khối Cubes. ....	133
6.3.2.	Thực hiện trên SQL. ....	134
6.3.3.	Thực hiện trên Excel. ....	135
6.3.4.	Thực hiện trên PowerBI. ....	137
6.4.	Hiển thị mức độ nghiêm trọng trung bình từng năm của quận NEW YORK với loại thời tiết là Cloudy. ....	139
6.4.1.	Thực hiện trên các khối Cubes. ....	139
6.4.2.	Thực hiện trên SQL. ....	140
6.4.3.	Thực hiện trên Excel. ....	140
6.4.4.	Thực hiện trên PowerBI. ....	142
6.5.	Hiển thị mức độ nghiêm trọng trung bình của tai nạn xảy ra từng tháng của từng năm. 144	
6.5.1.	Thực hiện trên các khối Cubes. ....	144
6.5.2.	Thực hiện trên SQL. ....	145
6.5.3.	Thực hiện trên Excel. ....	145
6.5.4.	Thực hiện trên PowerBI. ....	147
6.6.	Hiển thị tổng khoảng cách và số vụ tai nạn xảy ra từng tháng của từng năm với loại thời tiết là FAIR. ....	148
6.6.1.	Thực hiện trên các khối Cubes. ....	148
6.6.2.	Thực hiện trên SQL. ....	148
6.6.3.	Thực hiện trên Excel. ....	149
6.6.4.	Thực hiện trên PowerBI. ....	151
6.7.	Liệt kê các quận và quận có số vụ tai nạn từ 3000 trở lên. ....	153
6.7.1.	Thực hiện trên các khối Cubes. ....	153
6.7.2.	Thực hiện trên SQL. ....	154
6.7.3.	Thực hiện trên Excel. ....	155
6.7.4.	Thực hiện trên PowerBI. ....	158
6.8.	Tổng số vụ tai nạn theo quý của từng năm (DRILLDOWNLEVEL). ....	161
6.8.1.	Thực hiện trên các khối Cubes. ....	161
6.8.2.	Thực hiện trên SQL. ....	161

## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

6.8.3. Thực hiện trên Excel .....	162
6.8.4. Thực hiện trên PowerBI .....	164
6.9. Top 4 tháng của 2 năm có số lượng tai nạn cao nhất theo bang (Hàm GENERATE với TOP COUNT).....	165
6.9.1. Thực hiện trên các khối Cubes. ....	165
6.9.2. Thực hiện trên SQL .....	166
6.9.3. Thực hiện trên Excel. ....	167
6.9.4. Thực hiện trên PowerBI. ....	169
6.10. Số vụ tai nạn theo từng năm của các tiểu bang ngoài trừ bang NY và bang NV. 172	
6.10.1. Thực hiện trên các khối Cubes. ....	172
6.10.2. Thực hiện trên SQL. ....	173
6.10.3. Thực hiện trên Excel. ....	174
6.10.4. Thực hiện trên PowerBI. ....	177
<b>CHƯƠNG 4. QUÁ TRÌNH DATAMINING .....</b>	<b>180</b>
4.1. Tiền xử lý dữ liệu .....	180
4.1.1. Bổ sung tính năng.....	180
4.1.2. Kiểm tra mối tương quan giữa các tính năng .....	181
4.1.3. Lựa chọn tính năng .....	182
4.1.4. Xóa trùng lắp .....	182
4.1.5. Xử lý giá trị sai hoặc thiếu sót.....	183
4.1.6. Kiểm tra phương sai của tính năng.....	188
4.1.7. Xử lý dữ liệu không cân bằng .....	189
4.1.8. Chuẩn hóa tính năng .....	190
4.1.9. Mã hóa tính năng .....	191
4.2. Ứng dụng mô hình thuật toán khai thác dữ liệu.....	194
4.2.1. Chia dữ liệu trước khi xây dựng mô hình thuật toán.....	195
4.2.2. Decision Tree.....	196
4.2.3. Random Forest.....	200
<b>CHƯƠNG 5. TÀI LIỆU THAM KHẢO .....</b>	<b>202</b>

## LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành đến Cô Nguyễn Thị Kim Phụng – giảng viên môn Kho dữ liệu và OLAP về sự hướng dẫn và sự hỗ trợ quý báu mà Cô đã dành cho chúng em trong quá trình thực hiện đồ án môn học "Xây dựng kho dữ liệu US ACCIDENTS".

Suốt quá trình học tập, em đã được tiếp cận với những kiến thức nền tảng quan trọng về kho dữ liệu và OLAP, bao gồm các khái niệm cơ bản, mô hình dữ liệu, kỹ thuật trích xuất, chuyển đổi và tải dữ liệu, các công cụ và kỹ thuật phân tích dữ liệu. Nhờ sự giảng dạy cẩn thận và giải thích chi tiết của cô, em đã có thể nắm bắt kiến thức một cách hiệu quả và áp dụng vào thực tế trong đề tài xây dựng kho dữ liệu US ACCIDENTS. Nếu không có những lời hướng dẫn của cô thì nhóm chúng em nghĩ đồ án này của nhóm rất khó có thể hoàn thiện được. Một lần nữa, chúng em xin bày tỏ lòng biết ơn chân thành đến Cô.

Dựa trên những kiến thức được Cô cung cấp trên trường kết hợp với việc tự mày mò, tìm hiểu những công cụ và kiến thức mới, nhóm đã cố gắng thực hiện đồ án một cách tốt nhất. Trong thời gian thực hiện đề tài, nhóm chúng em đã vận dụng những kiến thức nền tảng mà trên trường Cô đã truyền đạt và kết hợp việc học hỏi và nghiên cứu. Từ đó, nhóm chúng em đã vận dụng được tối đa những gì đã được tiếp thu để hoàn thành báo cáo đồ án một cách tốt nhất. Tuy nhiên, trong quá trình thực hiện, nhóm chúng em cũng không tránh khỏi những thiếu sót. Chính vì vậy, nhóm chúng em rất mong được nghe những lời góp ý từ phía Cô nhằm hoàn thiện đồ án này hơn để học tập thêm kiến thức mới và là hành trang để nhóm chúng em thực hiện tiếp các đề tài khác trong tương lai.

Lời cuối, nhóm chúng em xin kính chúc cô thậ dồi dào sức khỏe, vững tin và thành công trên con đường truyền đạt kiến thức cho các bạn sinh viên.

Xin chân thành cảm ơn Cô!  
Nhóm thực hiện

## NHẬN XÉT CỦA GIÁO VIÊN



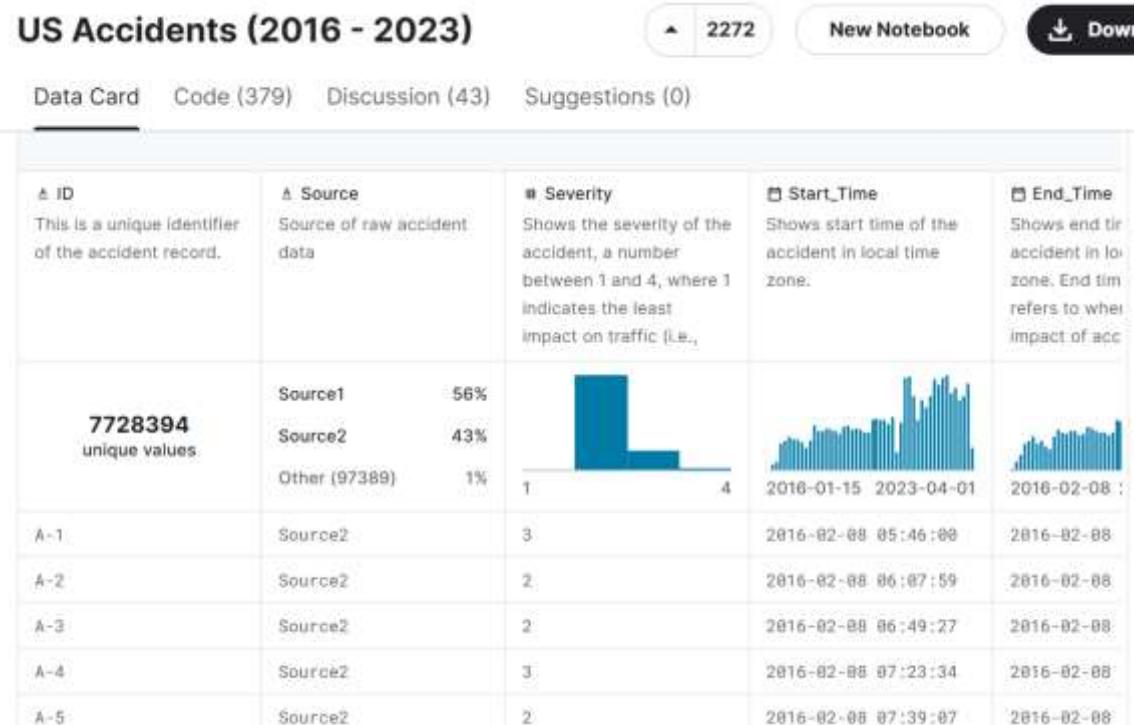
## CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ DỮ LIỆU

### 1. PHÁT BIỂU VỀ DỮ LIỆU

#### 1.1. Mô tả về dữ liệu

- Kho dữ liệu US Accidents là một kho dữ liệu thu nhập các vụ tai nạn ở US trong vòng 7 năm (từ năm 2016 – 2023).
- Thông qua kho dữ liệu người dùng có thể biết được thông tin ngày xảy ra tai nạn, tuyến đường xảy ra, thành phố, nhiệt độ, thời tiết, ngày/ đêm, mức gió, ...
- Kho dữ liệu được xây dựng với hướng chủ đề Geospatial analysis (Phân tích không gian địa lý).
- Kho dữ liệu gồm 7728394 dòng và 46 thuộc tính.
- Link dataset gốc: [Dataset US Accidents](#)

#### 1.2. Thuộc tính của kho dữ liệu



#### 1.3. Kho dữ liệu đã xử lý

- Sau khi lọc dữ liệu ta được 382660 dòng và 25 thuộc tính sử dụng việc phân tích cho đề tài

STT	Tên thuộc tính	Kiểu	Ý nghĩa của thuộc tính
-----	----------------	------	------------------------

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

1	ID	String	Số nhận dạng du nhất câu hỏi sơ vụ tai nạn
2	Source	String	Nguồn dữ liệu tai nạn thô
3	Severity	Int	Cho biết mức độ nghiêm trọng của vụ tai nạn. Có giá trị từ 1 đến 4, trong đó 1 cho biết tác động ít nhất đến giao thông (short delay).
4	Start_Time	DateTime	Hiển thị thời gian bắt đầu vụ tai nạn theo múi giờ địa phương (time_zone). Định dạng ngày: MM / DD / YYYY hh: mm.
5	Distance(mi)	Float	Chiều dài của đoạn đường bị ảnh hưởng bởi vụ tai nạn. (Đo bằng mile, trong đó 1 mile = 1,609344 km = 1.609,344 m)
6	Street	String	Hiển thị tên đường trong bản ghi địa chỉ.
7	City	String	Hiển thị thành phố trong bản ghi địa chỉ.
8	Country	String	Hiển thị quốc trong hồ sơ địa chỉ.
9	State	String	Hiển thị tiểu bang trong bản ghi địa chỉ.
10	Airport_Code	String	Biểu thị một trạm thời tiết ở sân bay và là trạm gần nhất với vị trí xảy ra tai nạn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

11	Temperature(F)	Float	Hiển thị nhiệt độ (tính bằng °F).
12	Visibility(mi)	Float	Cho biết khả năng nhìn xa (tính bằng dặm)
13	Wind_Direction	String	Hiển thị hướng gió. Các giá trị có thể có là: – Calm: gió lặng – E, East: có gió hướng đông – ENE, NNE, NNW, NW....
14	Wind_Speed(mph)	Float	Hiển thị tốc độ gió (tính bằng dặm / giờ).
15	Precipitation(in)	Float	Hiển thị lượng mưa (tính bằng inch), nếu có.
16	Weather_Condition	String	Hiển thị tình trạng thời tiết (mưa, tuyết, giông, sương mù, v.v.)
17	Bump	Boolean	Cho biết sự hiện diện của gờ giảm tốc ở một vị trí gần đó.
18	Crossing	Boolean	Cho biết sự hiện diện của lối qua đường cho người đi bộ ở một địa điểm gần đó.
19	Junction	Boolean	Cho biết sự hiện diện của đường giao nhau ở một vị trí gần đó.
20	Railway	Boolean	Cho biết sự hiện diện của một đường sắt ở gần đó
21	Roundabout	Boolean	Cho biết sự hiện diện của một bùng binh ở gần đó
22	Station	Boolean	Cho biết sự hiện diện của một trạm tàu điện ở gần đó.

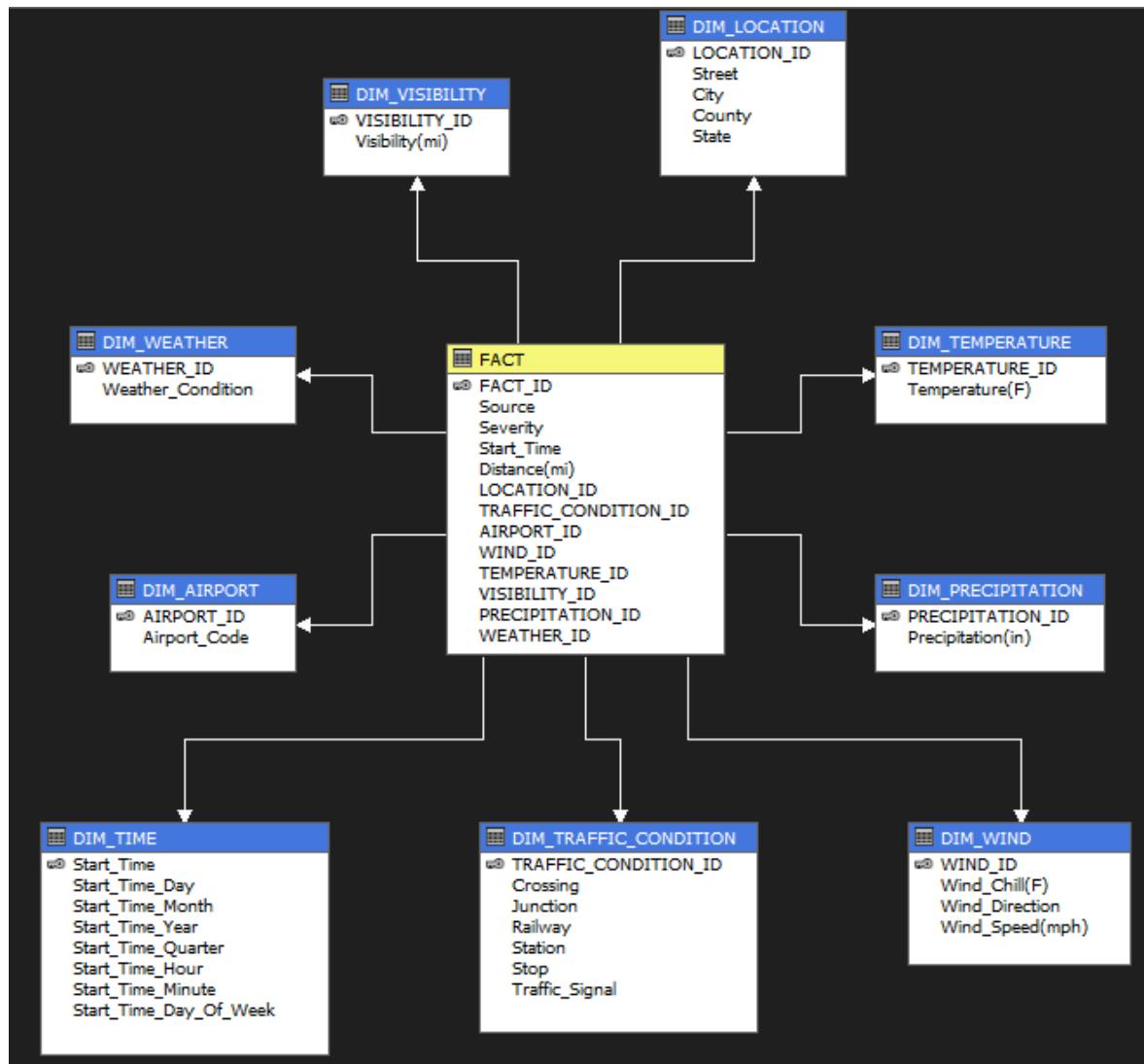
## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

23	Stop	Boolean	Cho biết sự hiện diện của một điểm dừng ở gần đó.
24	Traffic_Calming	Boolean	Cho biết sự hiện diện của các chướng ngại điều tiết giao thông đang ở gần đó. Bao gồm: đường gạch, lề đường rộng, hẻm, vòng xoay, tốc độ giới hạn thấp, đèn tín hiệu thông minh,... nhằm giảm tốc độ xe cộ, giúp tăng tính an toàn cho người đi bộ và người đi xe đạp, đồng thời giảm tiếng ồn và ô nhiễm khí thải ở đường phố đô thị.
25	Traffic_Signal	Boolean	Cho biết sự hiện diện của đèn tín hiệu giao thông ở vị trí gần đó.

# Đồ án xây dựng kho dữ liệu US ACCIDENTS

## 2. XÂY DỰNG KHO DỮ LIỆU

### 2.1. Sơ đồ hình sao minh họa



### 2.2. DIM\_TIME

- Có khóa chính là Start\_Time

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
🔑	Start_Time		DateTime	Hiển thị thời gian bắt đầu vụ tai nạn theo múi giờ địa phương (time_zone). Định dạng ngày: MM / DD / YYYY hh: mm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

	Start_Time_Day		int	Lấy ra ngày trong thời gian bắt đầu vụ tai nạn.
	Start_Time_Month		int	Lấy ra tháng trong thời gian bắt đầu vụ tai nạn.
	Start_Time_Year		int	Lấy ra năm trong thời gian bắt đầu vụ tai nạn.
	Start_Time_Quarter		int	Lấy ra quý trong thời gian bắt đầu vụ tai nạn.
	Start_Time_Hour		int	Lấy ra giờ trong thời gian bắt đầu vụ tai nạn.
	Start_Time_Minute		int	Lấy ra phút trong thời gian bắt đầu vụ tai nạn.
	Start_Time_Day_Of_Week		int	Lấy ra ngày trong tuần trong thời gian bắt đầu vụ tai nạn.

### 2.3. DIM\_LOCATION

- Có khóa chính là LOCATION\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	LOCATION_ID		Int	Mã địa điểm xảy ra vụ tai nạn.
	Street		String	Hiển thị tên đường trong bản ghi địa chỉ.
	City	x	String	Hiển thị thành phố trong bản ghi địa chỉ.
	County		String	Hiển thị quận trong hồ sơ địa chỉ.
	State		String	Hiển thị tiểu bang trong bản ghi địa chỉ.

### 2.4. DIM\_TRAFFIC\_CONDITION

- Có khóa chính là

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	TRAFFIC_CONDITION_ID		Int	Mã tình trạng giao thông nơi xảy ra tai nạn..
	Bump		Boolean	Cho biết sự hiện diện của gờ giảm tốc ở một vị trí gần đó.
	Crossing		Boolean	Cho biết sự hiện diện của lối qua đường cho người đi bộ ở một địa điểm gần đó.
	Junction		Boolean	Cho biết sự hiện diện của đường giao nhau ở một vị trí gần đó.
	Railway		Boolean	Cho biết sự hiện diện của một đường sắt ở gần đó
	Roundabout		Boolean	Cho biết sự hiện diện của một bùng binh ở gần đó
	Station		Boolean	Cho biết sự hiện diện của một trạm ga điện ở gần đó.
	Stop		Boolean	Cho biết sự hiện diện của một điểm dừng ở gần đó.
	Traffic_Calming		Boolean	Cho biết sự hiện diện của điều tiết giao thông đang ở gần đó.
	Traffic_Signal		Boolean	Cho biết sự hiện diện của đèn tín hiệu giao thông ở vị trí gần đó.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

### 2.5. DIM\_AIRPORT

- Có khóa chính là AIRPORT\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	AIRPORT_ID		int	Mã bản ghi thời tiết do trạm thời tiết gần nhất với vị trí xảy ra tai nạn ghi lại.
	Airport_Code		String	Biểu thị một trạm thời tiết ở sân bay và là trạm gần nhất với vị trí xảy ra tai nạn.

### 2.6. DIM\_WIND

- Có khóa chính là WIND\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	WIND_ID		int	Mã bản ghi thông tin về gió.
	Wind_Direction		string	Hiển thị hướng gió. Các giá trị có thể có là: – Calm: gió lặng – E, East: có gió hướng đông – ENE, NNE, NNW, NW....
	Wind_Speed(mph)		float	Hiển thị tốc độ gió (tính bằng dặm / giờ).

### 2.7. DIM\_TEMPERATURE

- Có khóa chính là TEMPERATURE\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	TEMPERATURE_ID		int	Mã bản ghi thông tin nhiệt độ.
	Temperature(F)		float	Hiển thị nhiệt độ (tính bằng °F).

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

### 2.8. DIM\_VISIBILITY

- Có khóa chính là VISIBILITY\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	VISIBILITY_ID		Int	Mã bản ghi thông tin tầm nhìn xa.
	Visibility		float	Cho biết khả năng nhìn xa (tính bằng dặm)

### 2.9. DIM\_PRECIPITATION

- Có khóa chính là PRECIPITATION\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	PRECIPITATION_ID		int	Mã của bảng sự kiện
	Precipitation(in)		float	Hiển thị lượng mưa (tính bằng inch), nếu có.

### 2.10. DIM\_WEATHER

- Có khóa chính là WEATHER\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	WEATHER_ID		int	Mã bản ghi thông tin tình trạng thời tiết
	Weather_Condition		float	Hiển thị tình trạng thời tiết (mưa, tuyết, giông, sương mù, v.v.)

### 2.11. FACT

- Có khóa chính là FACT\_ID

Khóa chính	Tên thuộc tính	NULL	Kiểu dữ liệu	Mô tả thuộc tính
	FACT_ID		String	Số nhận dạng duy nhất của hò sơ vụ tai nạn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

	Start_Time		DateTime	Thời gian bắt đầu vụ tai nạn.
	LOCATION_ID		Int	Mã địa điểm xảy ra vụ tai nạn.
	AIRPORT_ID		Int	Mã bản ghi thời tiết do trạm thời tiết gần nhất với vị trí xảy ra tai nạn ghi lại.
	TRAFFIC_CONDITION_ID		Int	thời tiết gần nhất với vị trí
	WEATHER_ID		Int	Mã bảng ghi thông tin về điều kiện thời tiết
	WIND_ID		Int	Mã bản ghi thông tin về gió
	TEAMPERATURE_ID		Int	Mã bản ghi thông tin nhiệt độ
	VISIBILIT_ID		Int	Mã bản ghi thông tin về tầm nhìn xa
	PRECIPITATION_ID		Int	Mã bản ghi thông tin lượng mưa
	Severity		Int	Mức độ nghiêm trọng của vụ tai nạn
	Distance(m)		Int	Chiều dài của đoạn đường bị ảnh hưởng bởi vụ tai nạn.

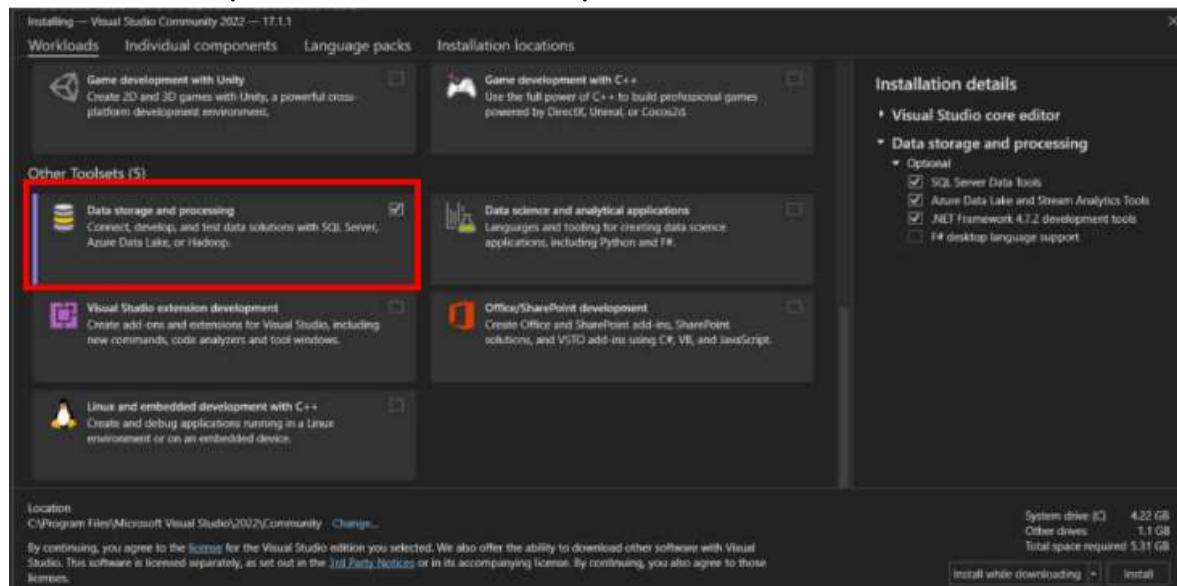
## CHƯƠNG 2. TÍCH HỢP DỮ LIỆU VÀ KHO (SSIS)

### 1. CHUẨN BỊ CÔNG CỤ VÀ DATA WAREHOUSE

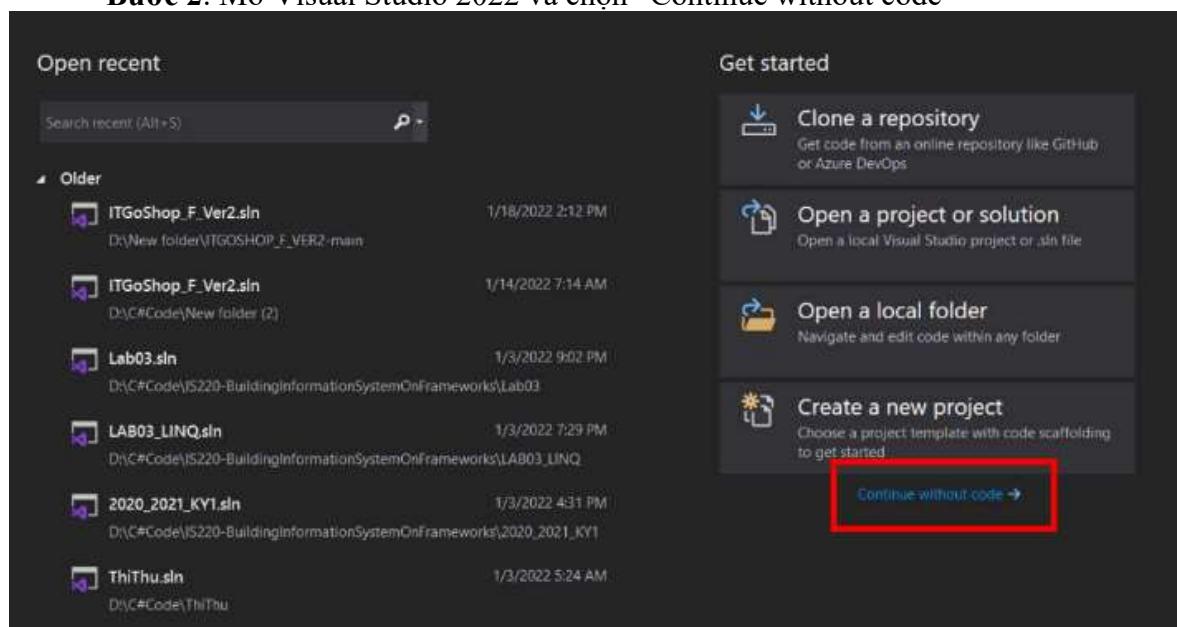
- Để thực hiện được quá trình SSIS ta cần chuẩn bị và cài đặt các công cụ sau:
  - Visual Studio Community 2022
  - SQL Server Integration Services Project
- **Bước 1:** Tải Visual Studio Community 2022 về máy. Trong lúc cài đặt,

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

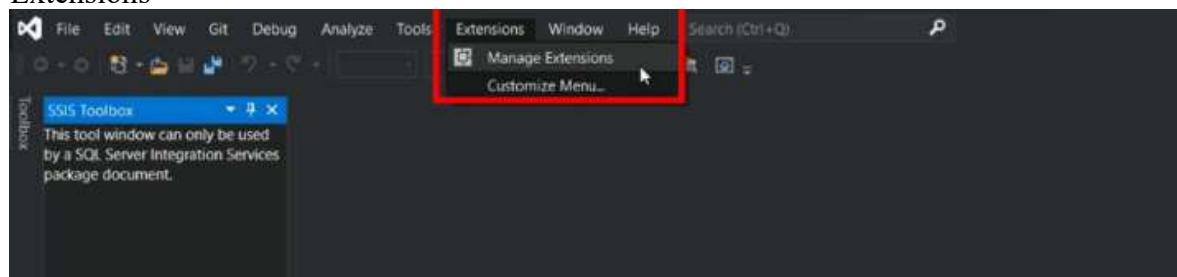
chọn mục “Data storage and processing” để cài đặt SQL Server Data Tools. Sau đó chọn Install để tiến hành cài đặt



- **Bước 2:** Mở Visual Studio 2022 và chọn “Continue without code”

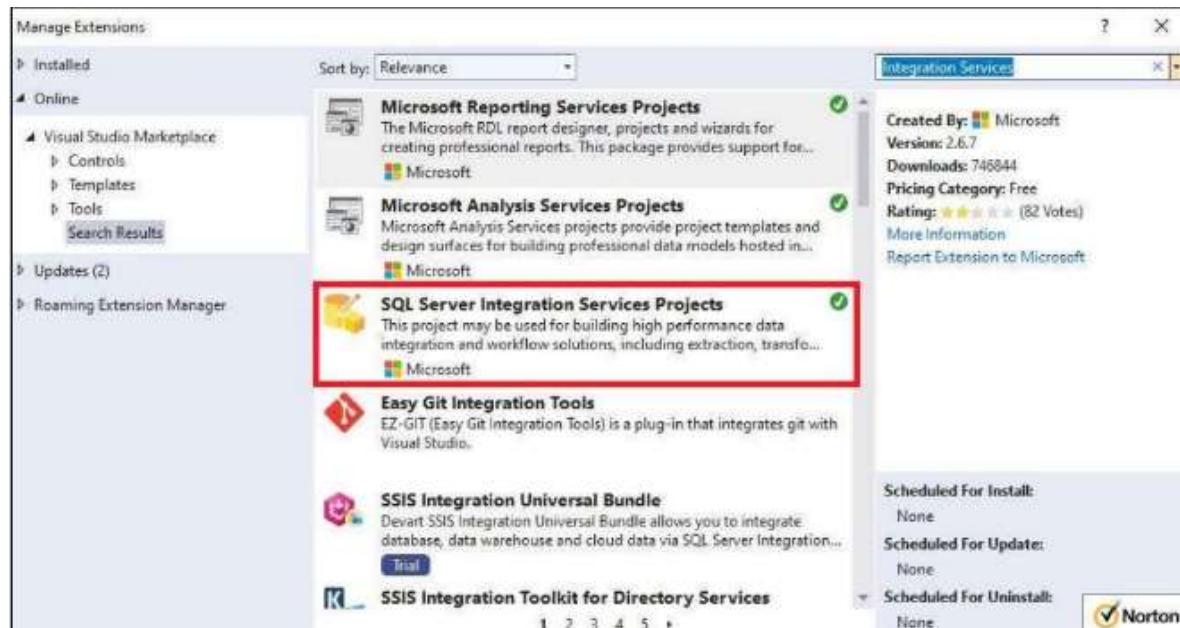


- **Bước 3:** Trong giao diện chính, click chọn “Extensions” > “Manage Extensions”

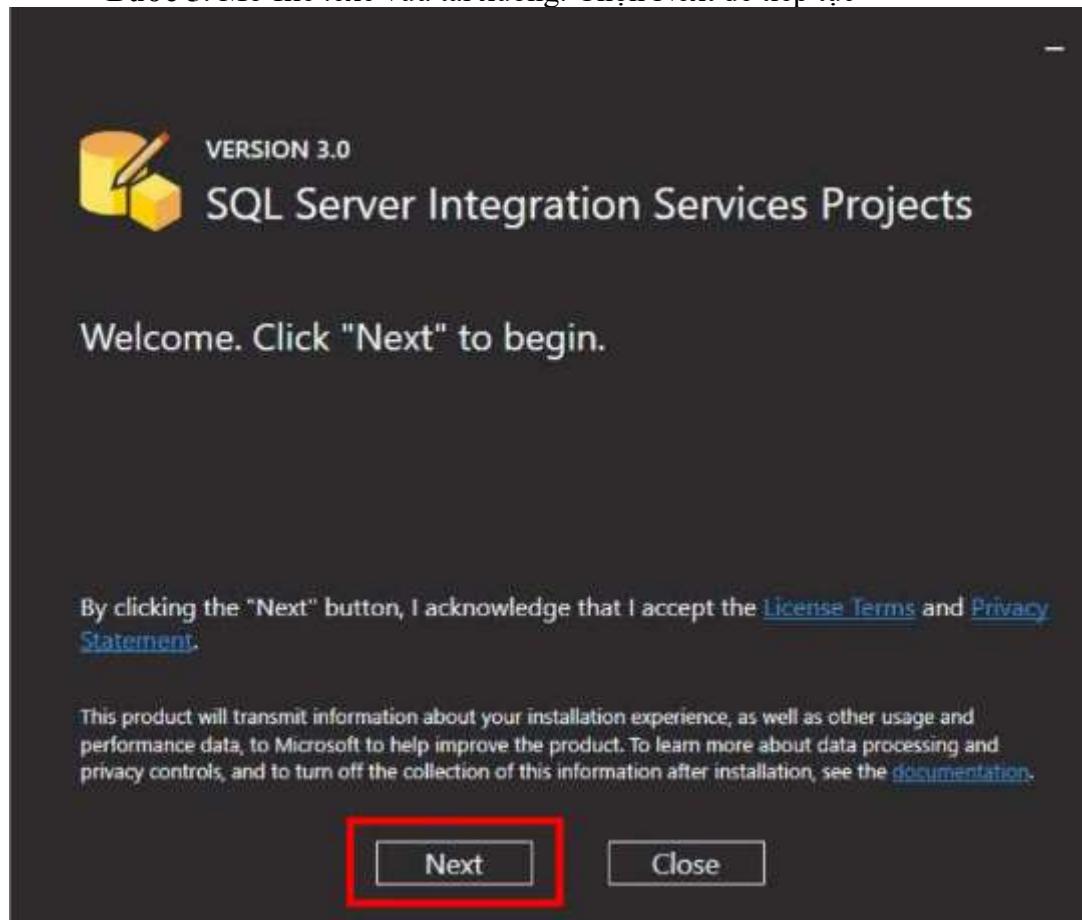


## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- **Bước 4:** Tìm và tải về công cụ SQL Server Integration Services Projects.



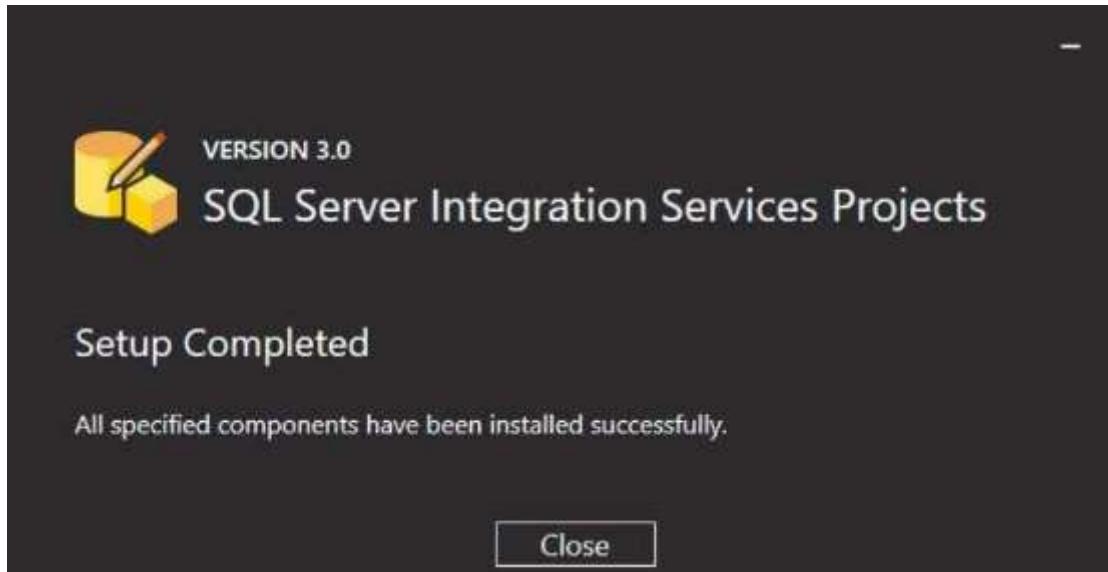
- **Bước 5:** Mở file .exe vừa tải xuống. Chọn Next để tiếp tục



- **Bước 6:** Tick chọn Visual Studio 2022 và chọn Install

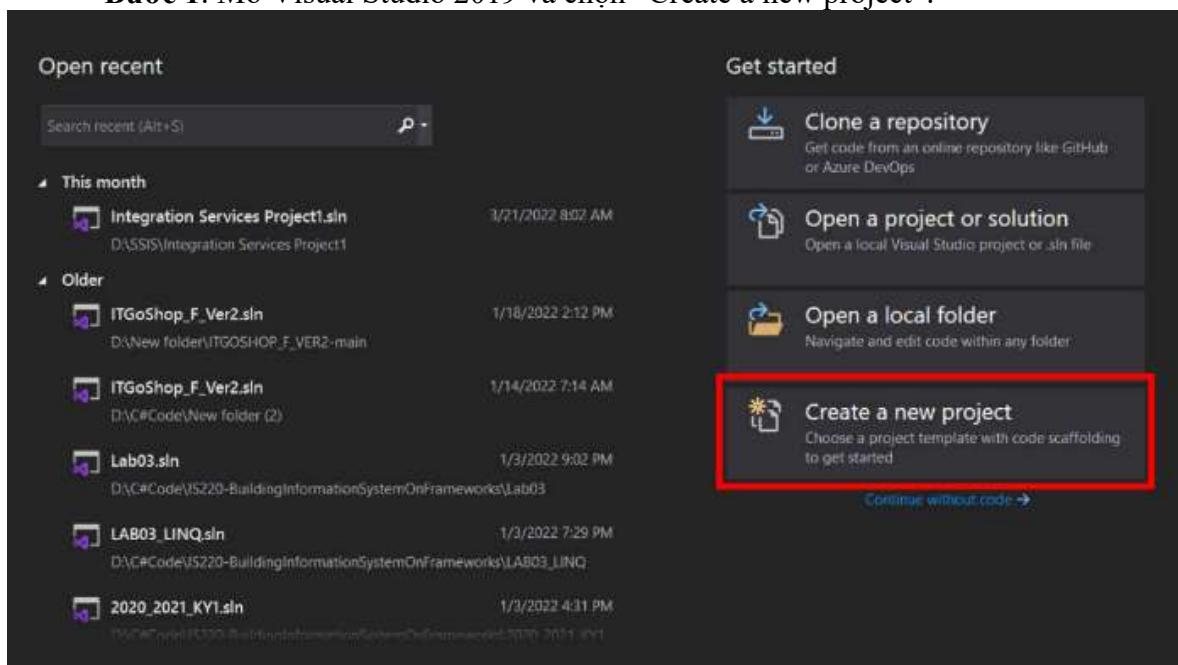
- Sau khi cài đặt thành công sẽ nhận được thông báo:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



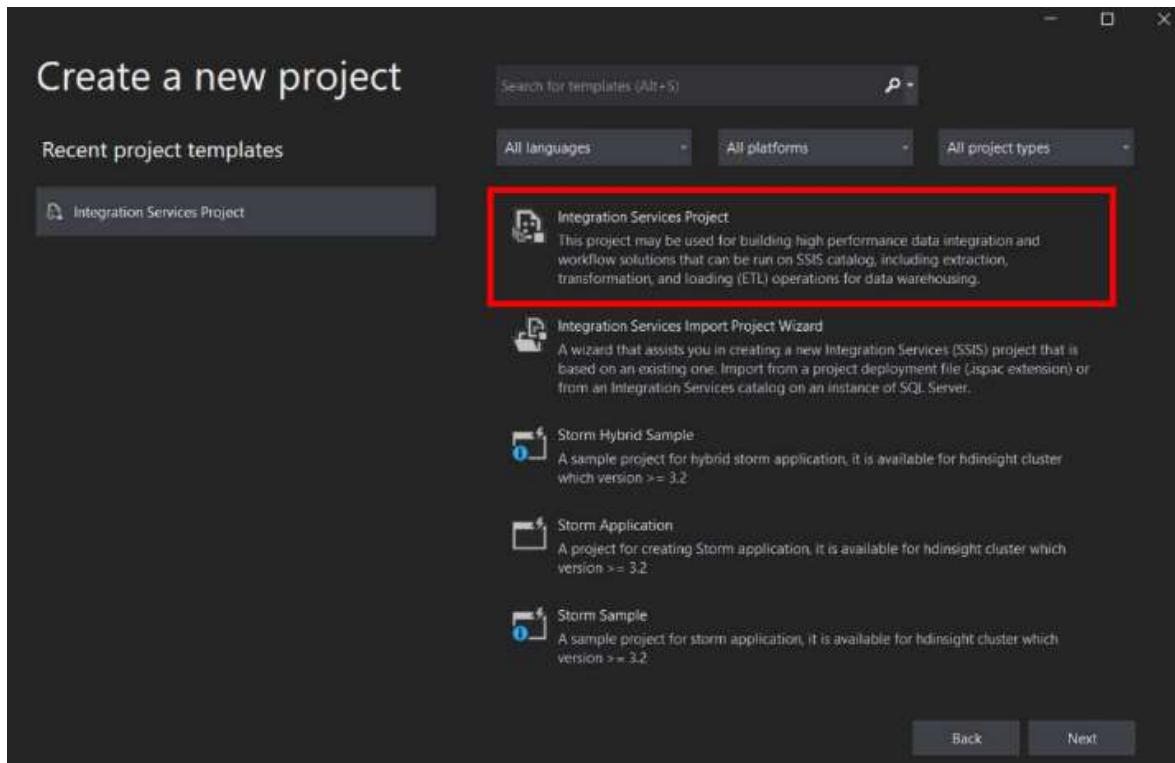
## 2. TẠO PROJECT SSIS TRONG VISUAL STUDIO 2022

- **Bước 1:** Mở Visual Studio 2019 và chọn “Create a new project”.

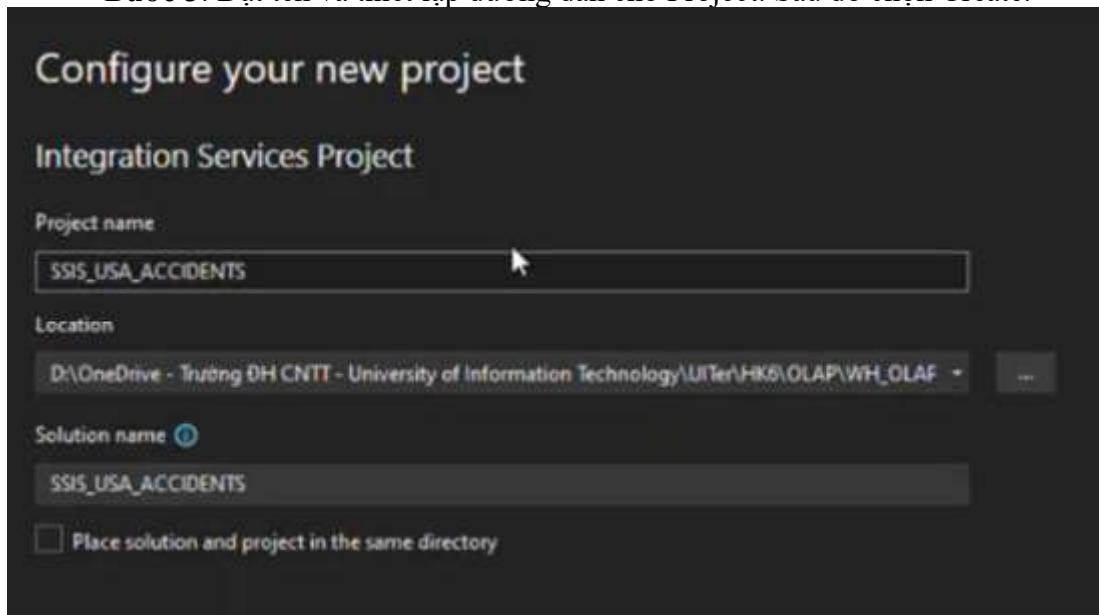


- **Bước 2:** Chọn Integration Services Project và chọn Next

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 3:** Đặt tên và thiết lập đường dẫn cho Project. Sau đó chọn Create.



### 3. TẠO BẢNG DIM VÀ BẢNG FACT

Trước khi tiến hành chia Dimension và bảng Fact, ta cần load dữ liệu gốc từ file .xlsx và Data Flow:

- **Bước 1:** Trong Data Flow, tạo một đối tượng Excel Source để lấy dữ liệu gốc từ file .xlsx. Chọn New để tạo một Excel Connection Manager
- **Bước 2:** Chọn nút Browse.. để tải lên file dữ liệu gốc lưu trong máy
- **Bước 3:** Chọn file dữ liệu .xlsx và chọn Open
- **Bước 4:** Xem lại các cột dữ liệu trong file dữ liệu đã được tải lên ở menu

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

### Columns

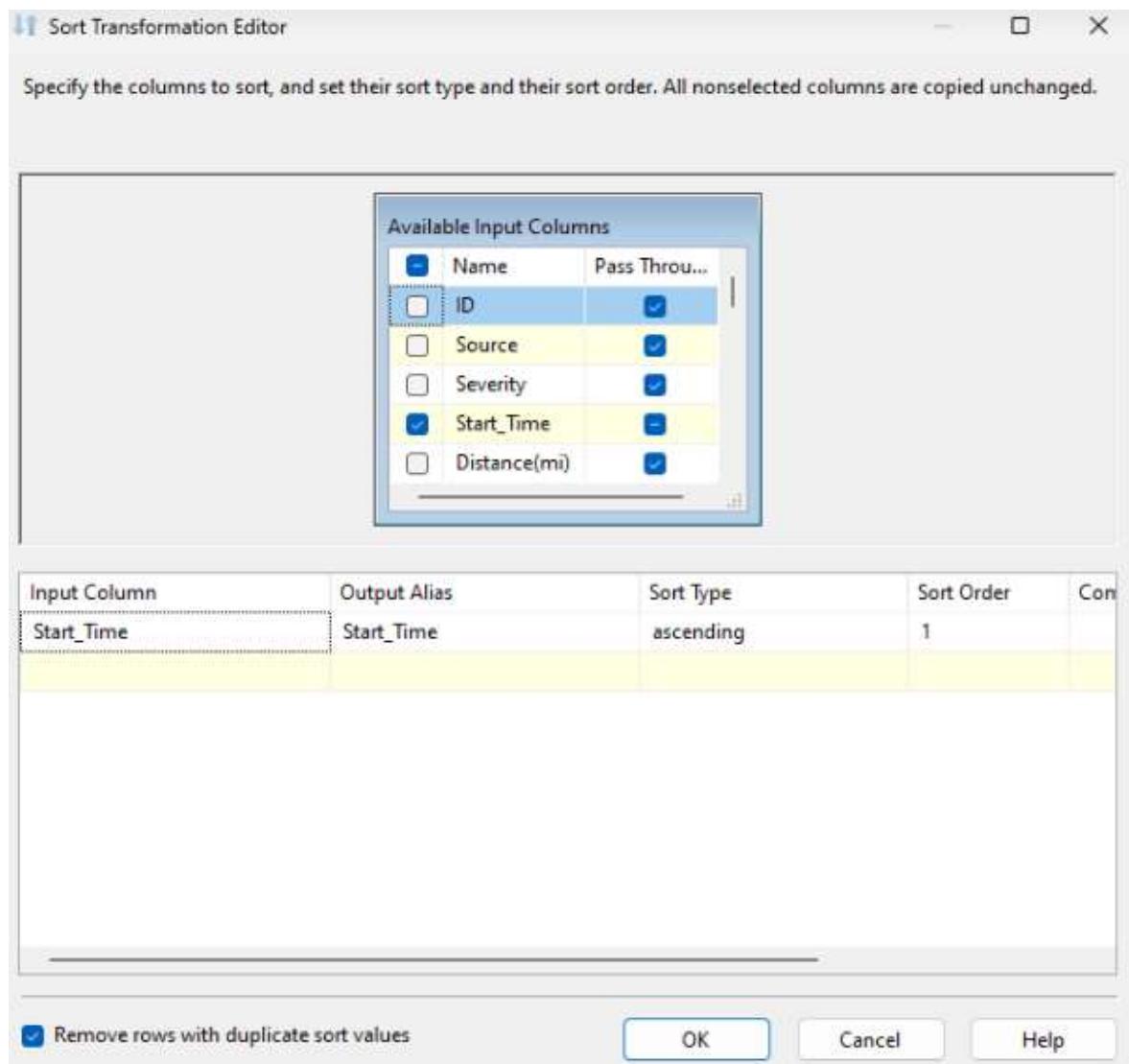
- **Bước 5:** Click chọn OK và kiểm tra lần nữa các cột dữ liệu ở dạng danh sách. Nhấn OK lần nữa để tiến hành hoàn tất quá trình load dữ liệu vào Excel Source
- **Bước 6:** Tạo Multicast để phân tán dữ liệu từ Excel Source đến các Dimension. Tiến hành kết nối Excel Source và Multicast.



### 3.1. Bảng DIM\_TIME

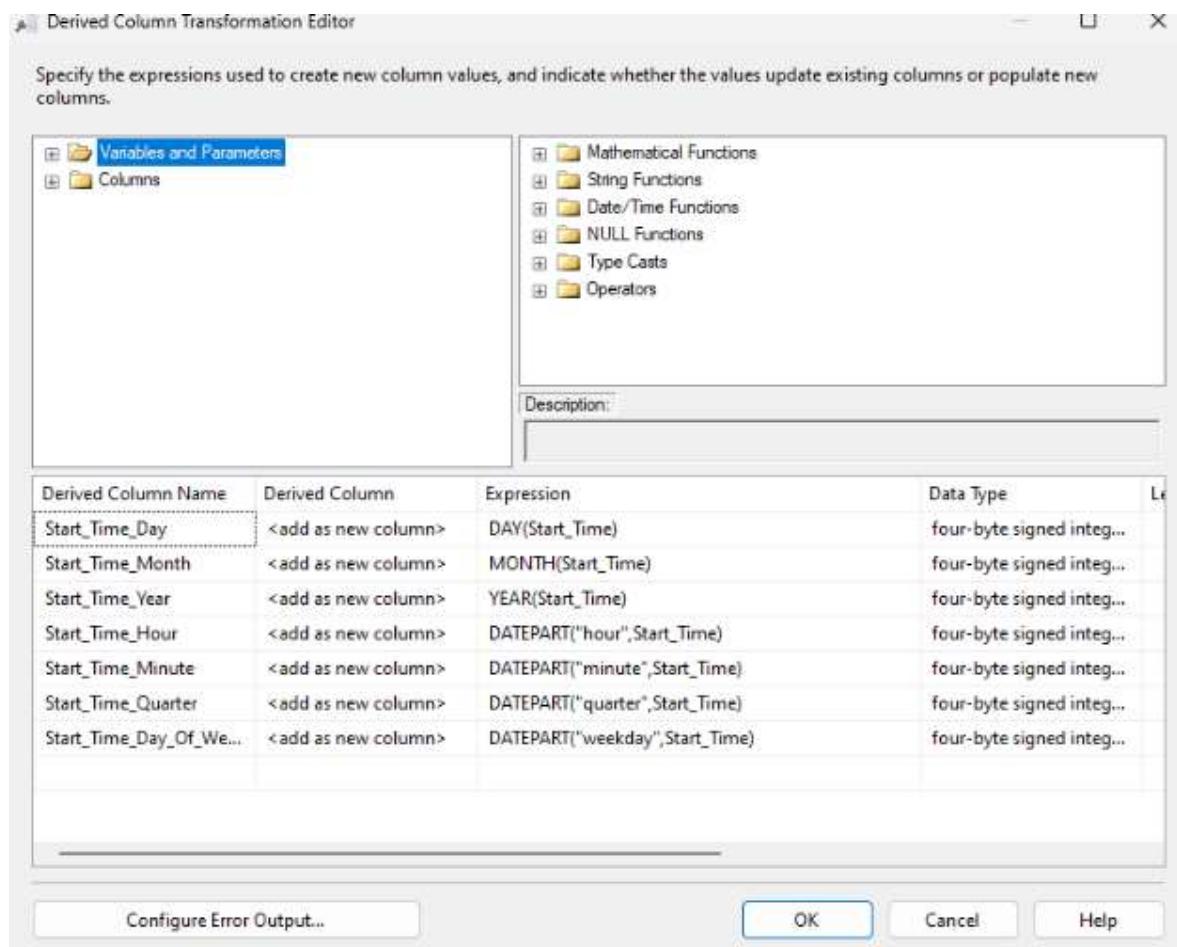
- Bước 1. Tạo mới một Sort có tên là Sort\_Dim\_Time để lấy ra các cột dữ liệu cần thiết cho Dim\_Time. Nhấn chuột phải và nhấp Edit để chọn Start\_Time làm cột dữ liệu cho Sort\_Dim\_Time.
  - Tick chọn Remove rows with duplicate sort values xóa các dòng dữ liệu trùng nhau, sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 2.** Thêm thành phần Derived Column và chọn Edit để chia cột dữ liệu
- **Bước 3.** Chia dữ liệu từ cột Start\_Time\_Date có kiểu dữ liệu dd/MM/yyyy HH:mm thành các cột Start\_Time\_Hour, Start\_Time\_Minute, Start\_Time\_Day, Start\_Time\_Month, Start\_Time\_Year, Start\_Time\_Quarter, Start\_Time\_Day\_of\_Week
  - Chọn phương thức DATEPART(<kiểu dữ liệu thời gian>, cột dữ liệu). Ở đây, ta chia Start\_Time\_Date thành cột Start\_Time\_Day nên ta cài đặt: DATEPART("day", [Start\_Time\_Date]).
- **Bước 4.** Tương tự chia Start\_Time\_Date thành các cột Start\_Time\_Hour, Start\_Time\_Minute, Start\_Time\_Month, Start\_Time\_Year, Start\_Time\_Quarter, Start\_Time\_Day\_of\_Week.
  - Tiếp theo, nhấn nút Configure Error Output...
- **Bước 5.** Ta thấy từ 1 cột Start\_Time\_Date được chia thành 7 cột ứng với lược đồ dữ liệu bảng Dim\_Time. Nhấn OK để hoàn tất quá trình chia cột.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



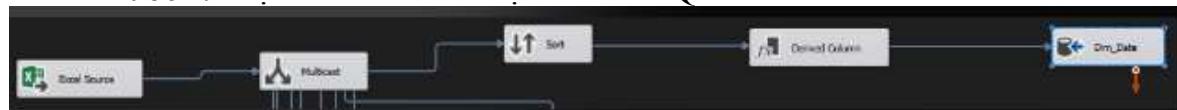
- **Bước 6.** Tạo Dim\_Time từ một OLE DB Destination. Double click vào OLE DB Destination này để tạo một connection mới đến MS SQL Server.

- Tiếp tục chọn New... để tạo một connection mới:

- Bước 6. Chọn tên server name trùng với server name MS SQL Server để ta có thể kết nối đến datawarehouse ACCIDENTS\_DW vừa tạo. Kết nối đến server bằng tài khoản window mặc định (Windows Authentication)

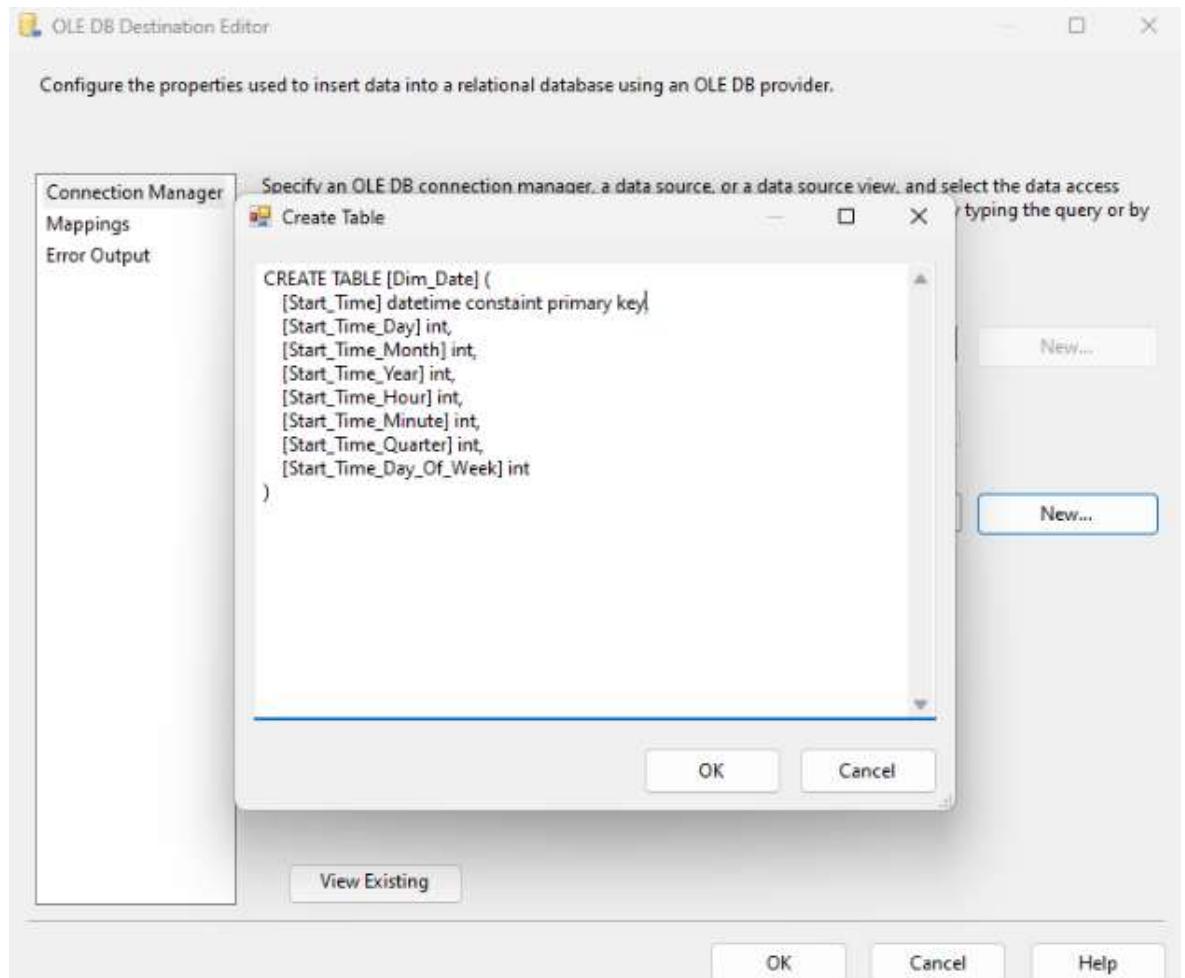
- Nhấn Test Connection để kiểm tra kết nối
- Ta tiến hành tạo bảng Dim\_Time ở MS SQL Server

- **Bước 7.** Chọn connection vừa tạo đến MS SQL Server và nhấn OK.



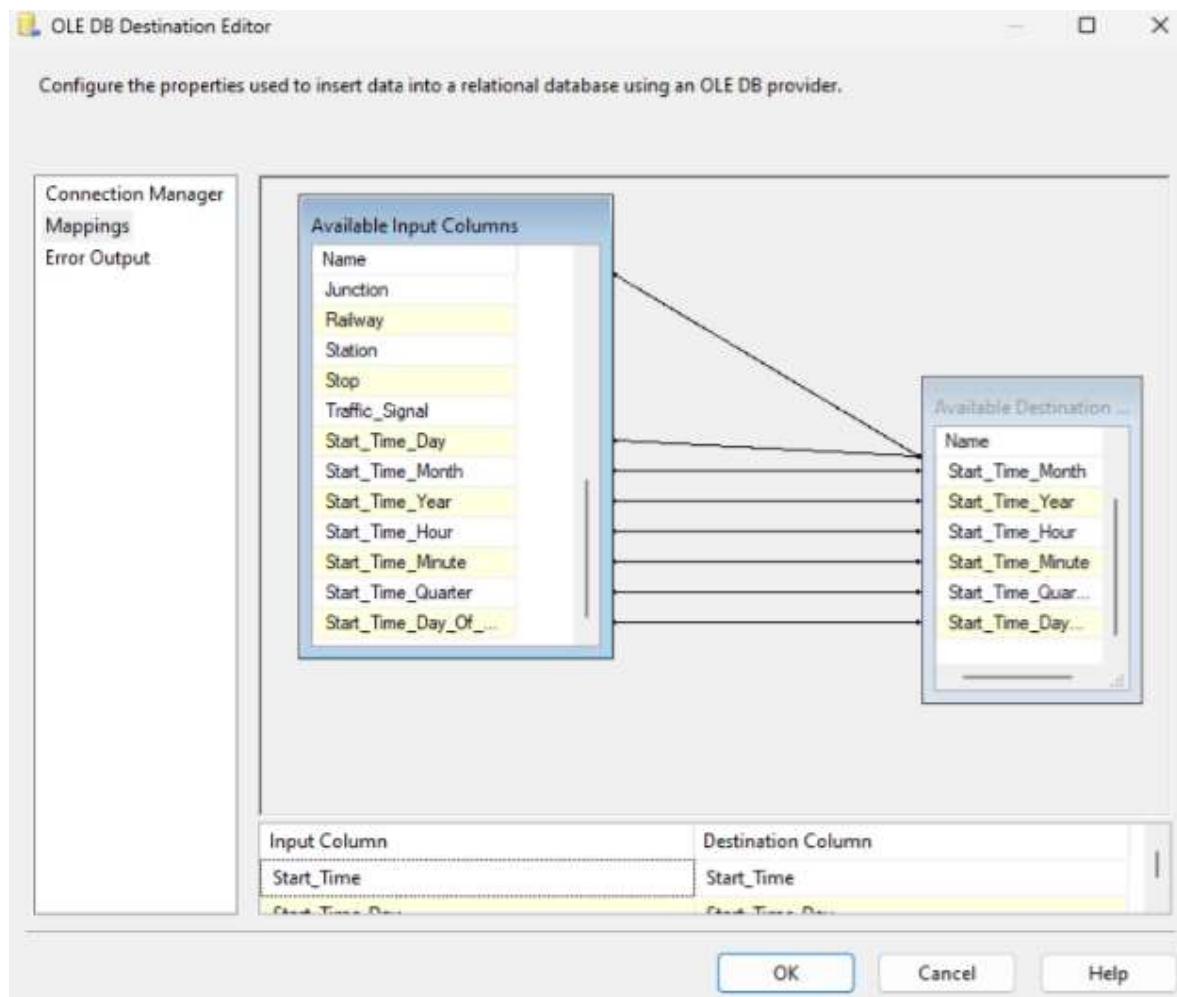
- **Bước 8.** Chọn New.. để tạo mới bảng

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 9.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

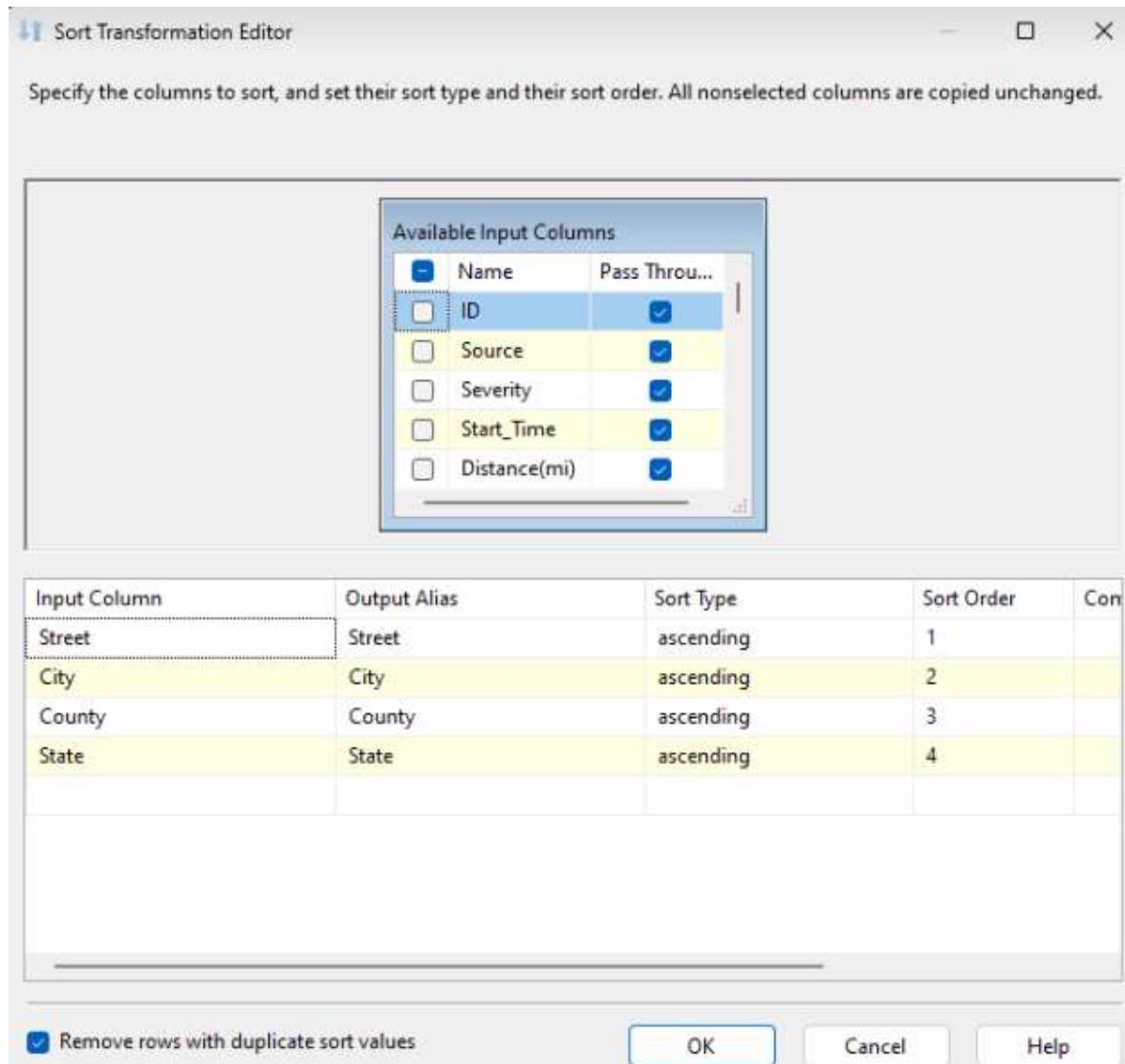


- Chọn OK để hoàn tất thiết lập.

### 3.2. Bảng DIM\_LOCATION

- Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Location cho Dim\_Location
- Bước 2.** Click chuột phải vào Sort\_Dim\_Location, chọn Edit: lần lượt chọn các cột Street, City, Country và State làm các cột để đổ dữ liệu vào Sort\_Dim\_Location
  - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

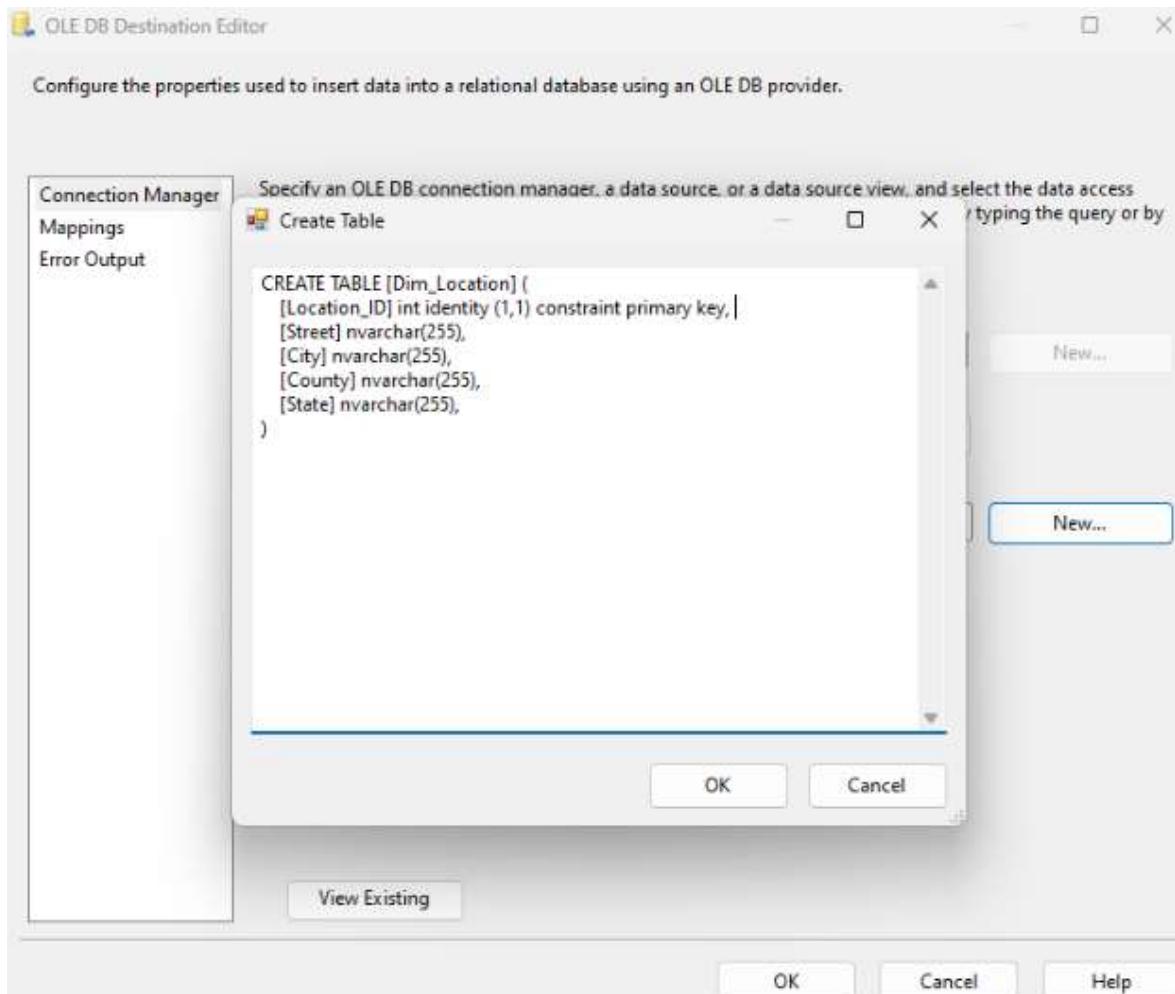


- **Bước 3.** Tạo mới một OLE DB Destination để đổ dữ liệu gốc sau khi đã được xử lý vào trong kho dữ liệu ACCIDENTS\_DW.

- **Bước 4.** Connection đến kho dữ liệu đã được tạo khi tạo Dim\_Time, vì vậy ta chỉ cần chọn New... để tạo bảng Dim\_Location



## Đồ án xây dựng kho dữ liệu US ACCIDENTS



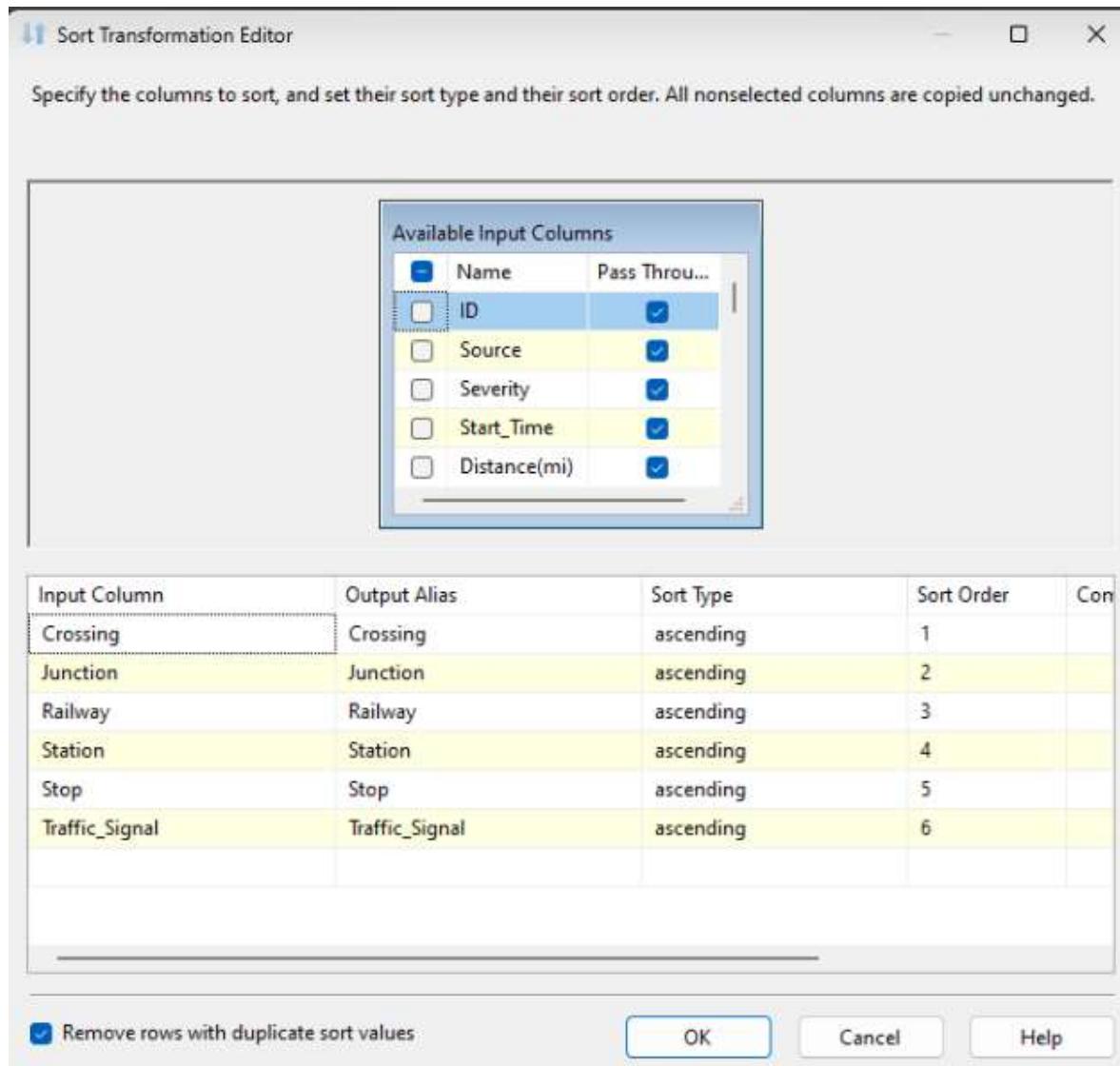
- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

- Chọn OK để hoàn tất thiết lập.

### 3.3. Bảng DIM\_TRAFFIC\_CONDITION

- **Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Traffic\_Location cho Dim\_Traffic\_Location
  - **Bước 2.** Click chuột phải vào Sort\_Dim\_Traffic\_Location, chọn Edit: lần lượt chọn các cột Bump, Crossing, Junction, Railway, Roundabout, Station, Stop, Traffic\_Calming và Traffic\_Signal làm các cột để đổ dữ liệu vào Sort\_Dim\_Traffic\_Location
    - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

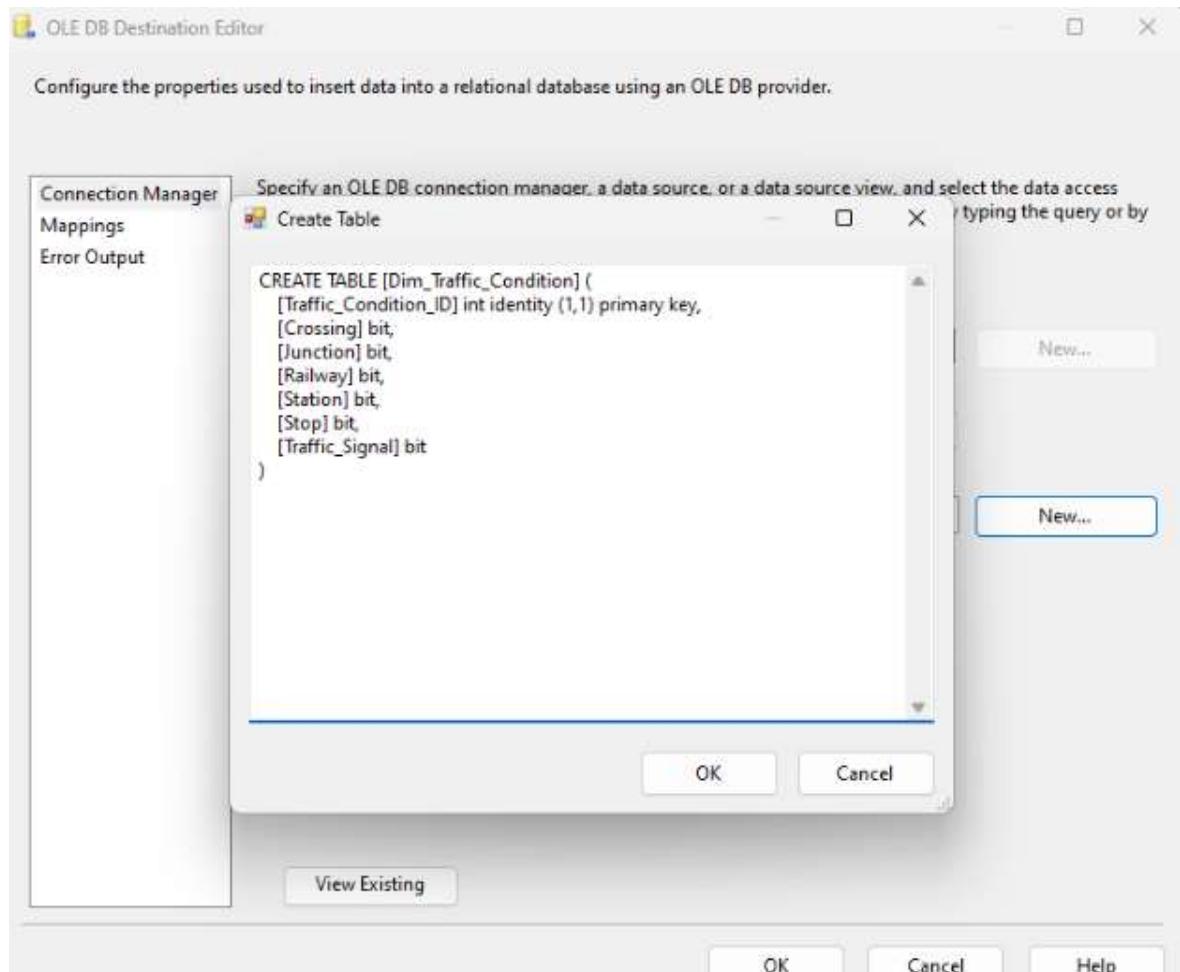


- **Bước 3.** Tạo mới một OLE DB Destination để đổ dữ liệu gốc sau khi đã được xử lý vào trong bảng Dim\_Traffic\_Condition kho dữ liệu ACCIDENTS\_DW.

- **Bước 4.** Connection đến kho dữ liệu đã được tạo khi tạo Dim\_Time, vì vậy ta chỉ cần chọn New... để tạo bảng Dim\_Traffic\_Condition



## Đồ án xây dựng kho dữ liệu US ACCIDENTS



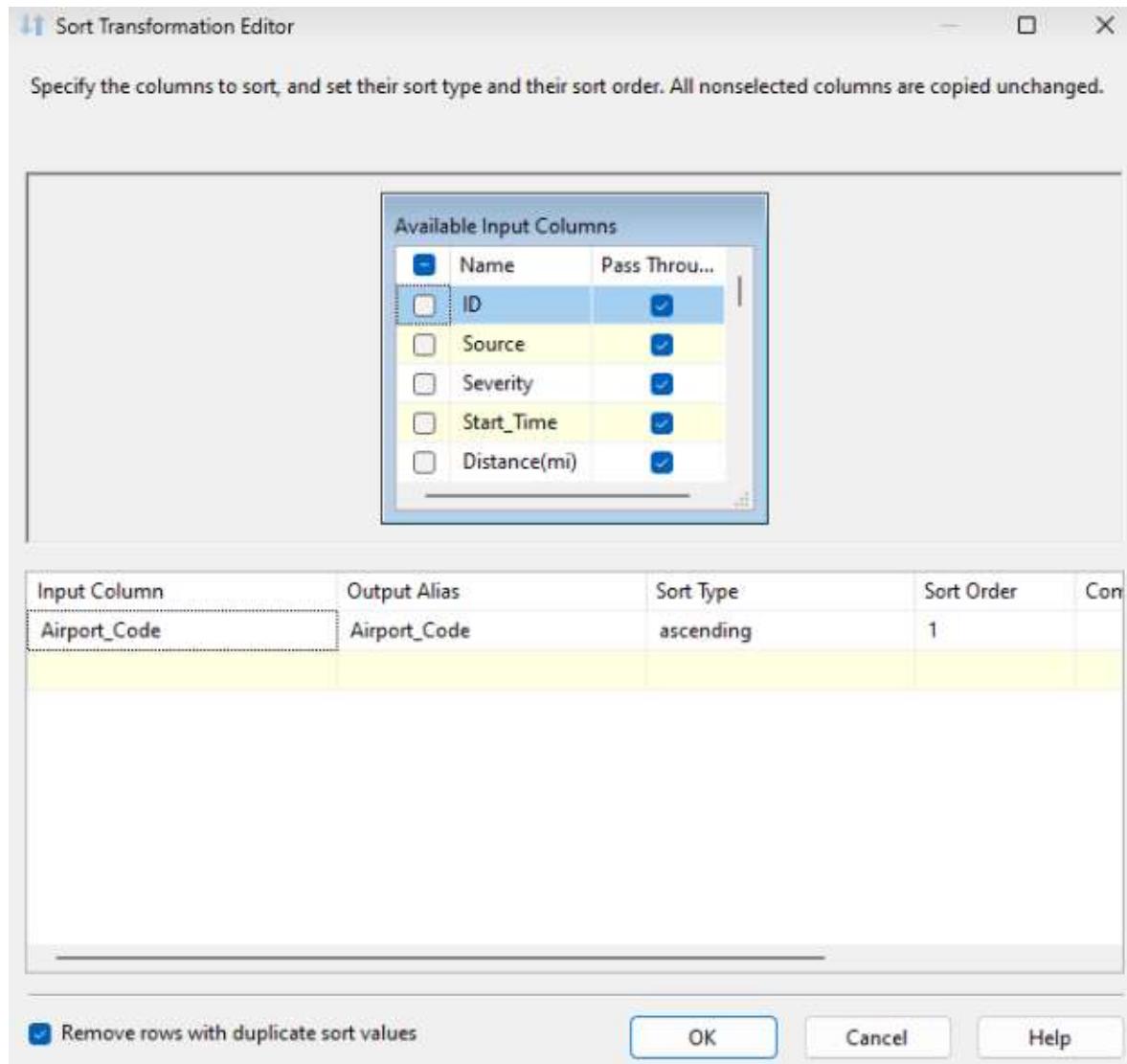
- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

- Chọn OK để hoàn tất thiết lập.

### 3.4. Bảng DIM\_AIRPORT

- **Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Airport cho Dim\_Airport
- **Bước 2.** Click chuột phải vào Sort\_Dim\_Airport, chọn Edit: lần lượt chọn các cột Street, City, Country và State làm các cột để đổ dữ liệu vào Sort\_Dim\_Airport.
  - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

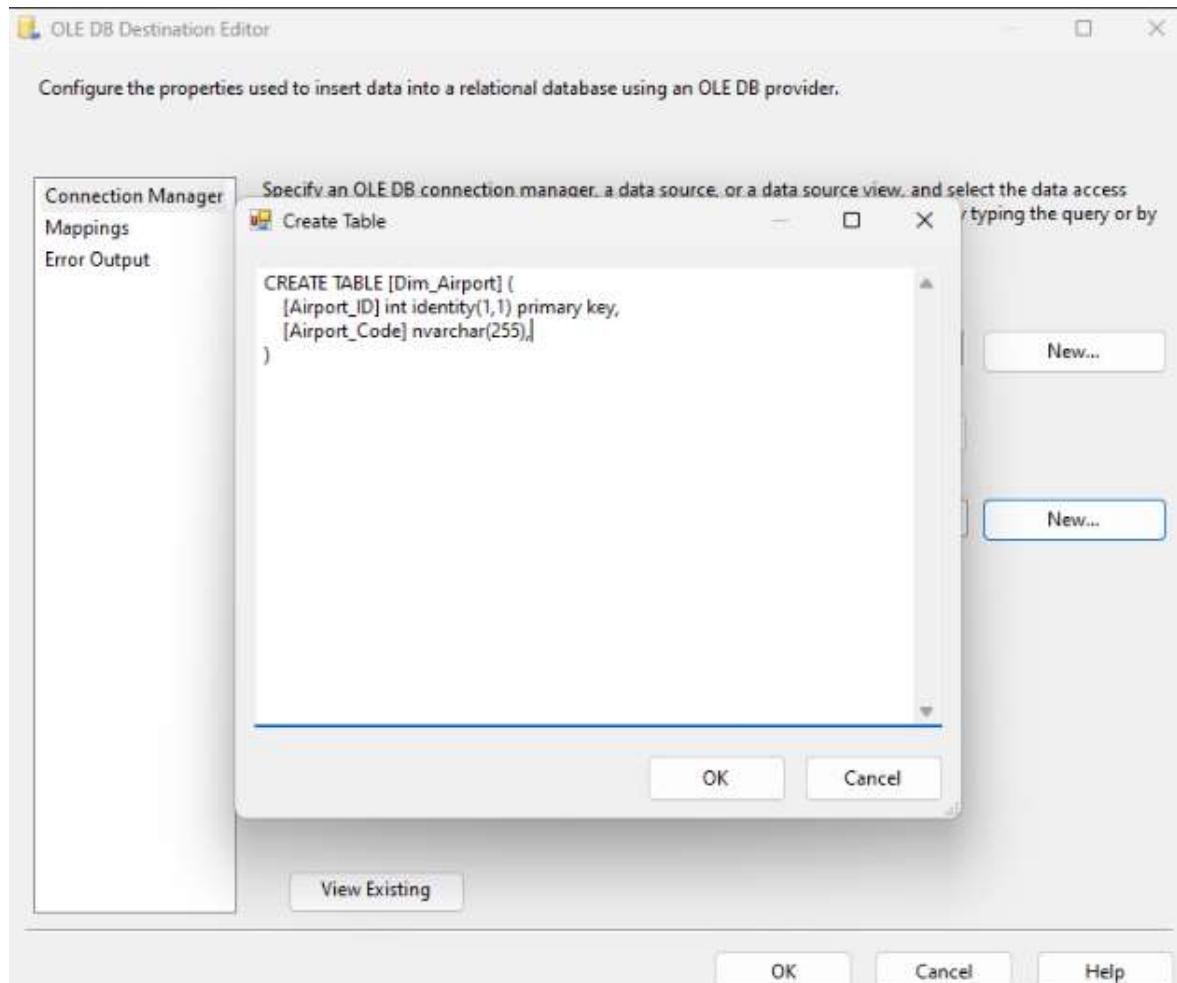


- **Bước 3.** Tạo mới một OLE DB Destination để đổ dữ liệu gốc sau khi đã được xử lý vào trong bảng Dim\_Airport kho dữ liệu ACCIDENTS\_DW.

- **Bước 4.** Connection đến kho dữ liệu đã được tạo khi tạo Dim\_Time, vì vậy ta chỉ cần chọn New... để tạo bảng Dim\_Airport



## Đồ án xây dựng kho dữ liệu US ACCIDENTS



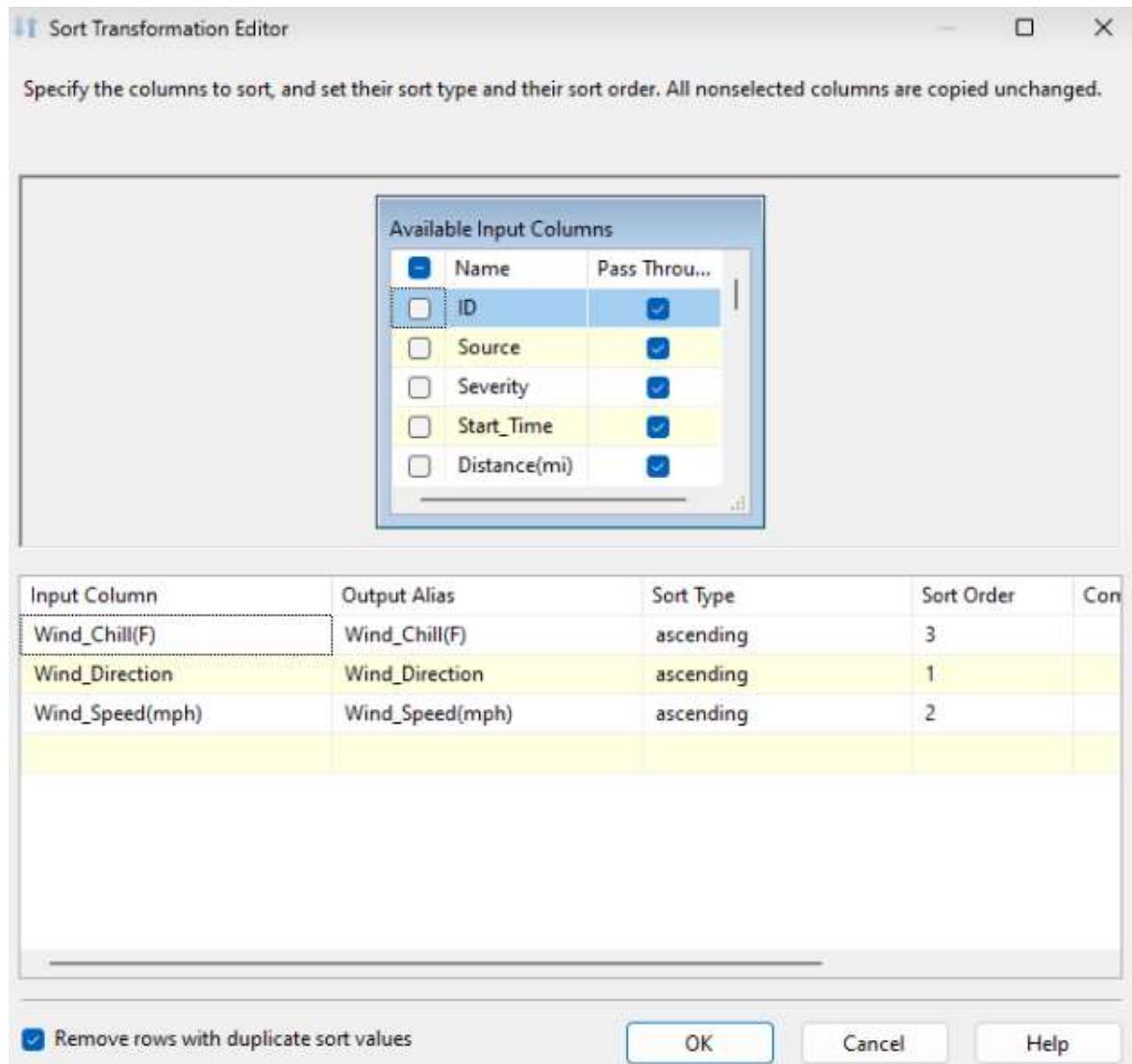
- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

- Chọn OK để hoàn tất thiết lập.

### 3.5. Bảng DIM\_WIND

- **Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Wind cho Dim\_Wind
- **Bước 2.** Click chuột phải vào Sort\_Dim\_Wind, chọn Edit: lần lượt chọn các cột Wind\_Direction và Wind\_Speed làm các cột để đổ dữ liệu vào Sort\_Dim\_Wind.
  - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

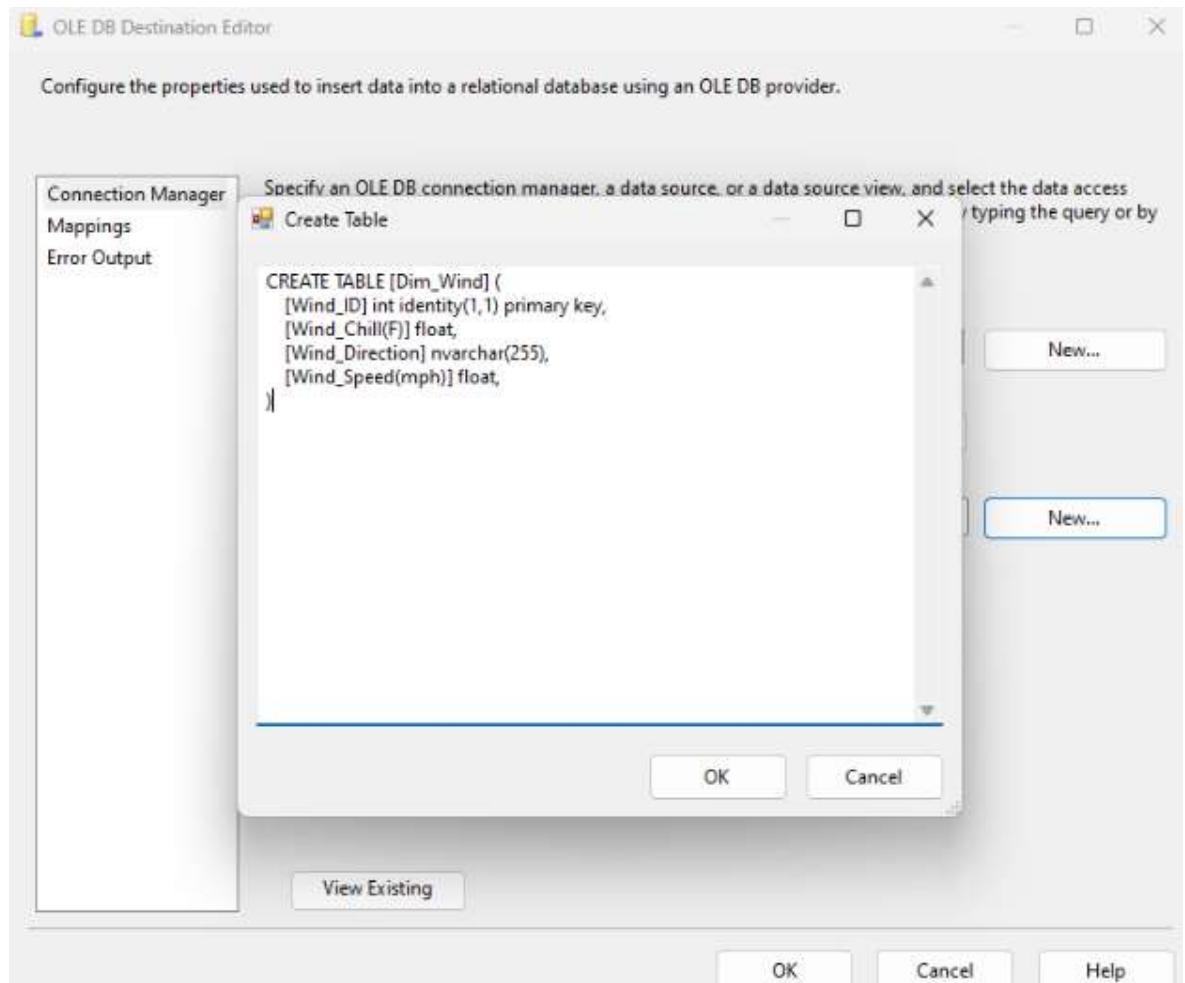


- **Bước 3.** Tạo mới một OLE DB Destination để đỗ dữ liệu gốc sau khi đã được xử lý vào trong bảng Dim\_Wind kho dữ liệu ACCIDENTS DW.



- **Bước 4.** Chọn New... để tạo bảng Dim\_Wind

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



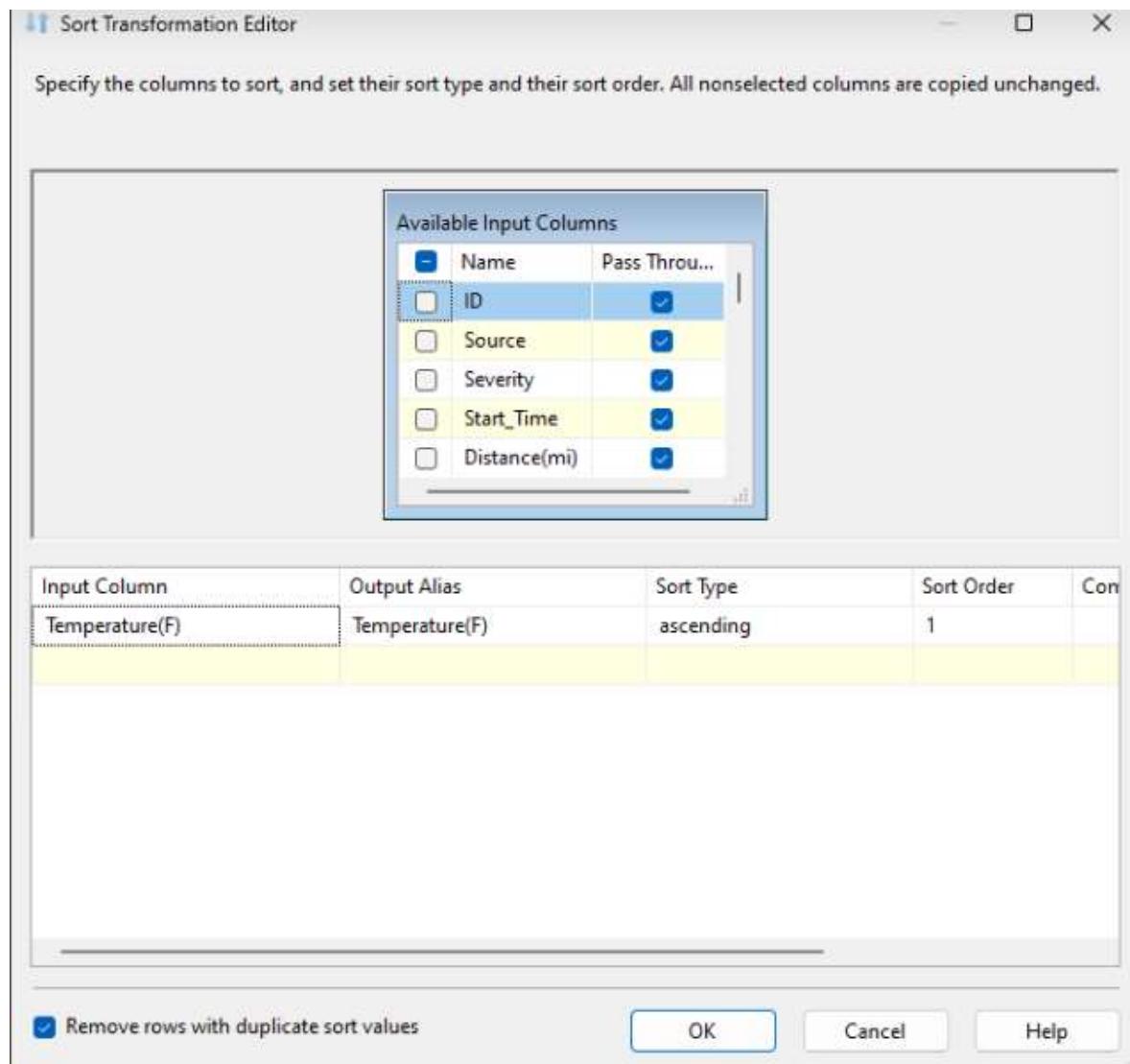
- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

- Chọn OK để hoàn tất thiết lập.

### 3.6. Bảng DIM\_TEMPERATURE

- **Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Temperature cho Dim\_Temperature
  - **Bước 2.** Click chuột phải vào Sort\_Dim\_Temperature, chọn Edit: lần lượt chọn các cột Temperature làm cột dữ liệu để đổ dữ liệu vào Sort\_Dim\_Temperature.
    - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

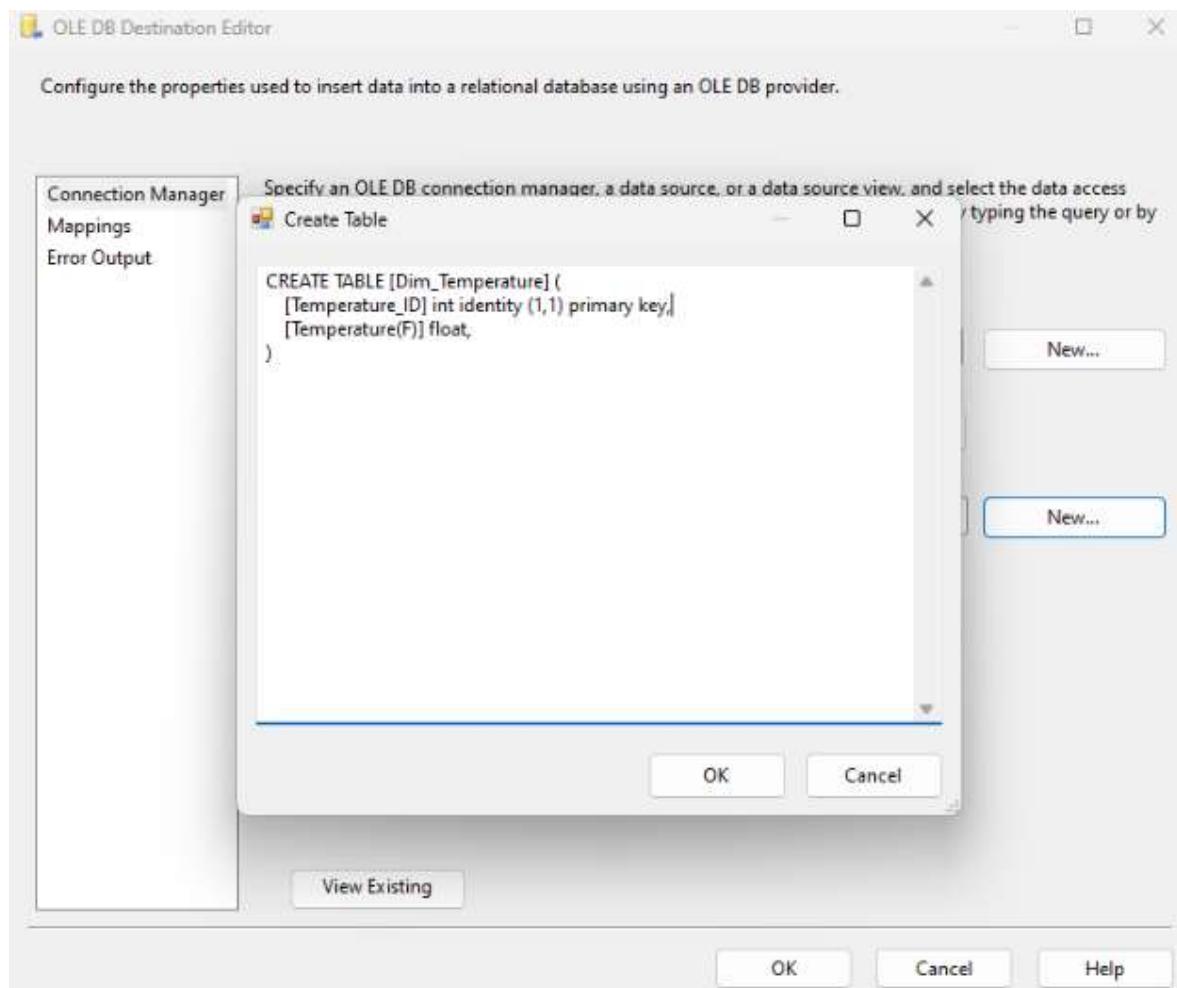


- **Bước 3.** Tạo mới một OLE DB Destination để đổ dữ liệu gốc sau khi đã được xử lý vào trong bảng Dim\_Temperature kho dữ liệu ACCIDENTS\_DW.



- **Bước 4.** Chọn New... để tạo bảng Dim\_Temperature

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



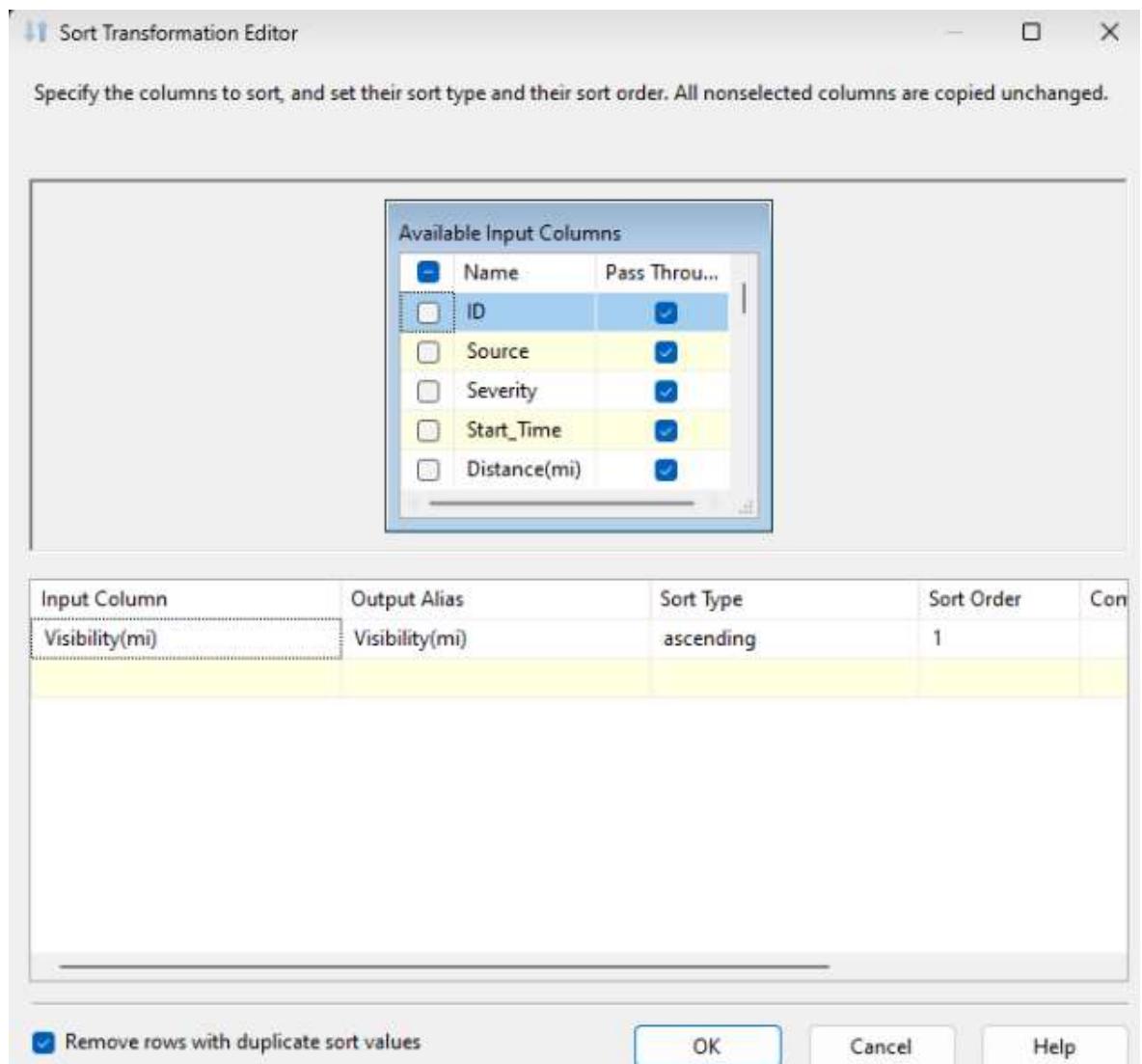
- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

- Chọn OK để hoàn tất thiết lập.

### 3.7. Bảng DIM\_VISIBILITY

- **Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Visibility cho Dim\_Visibility.
- **Bước 2.** Click chuột phải vào Sort\_Dim\_Visibility, chọn Edit: lần lượt chọn các cột Visibility làm cột dữ liệu để đổ dữ liệu vào Sort\_Dim\_Visibility.
  - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

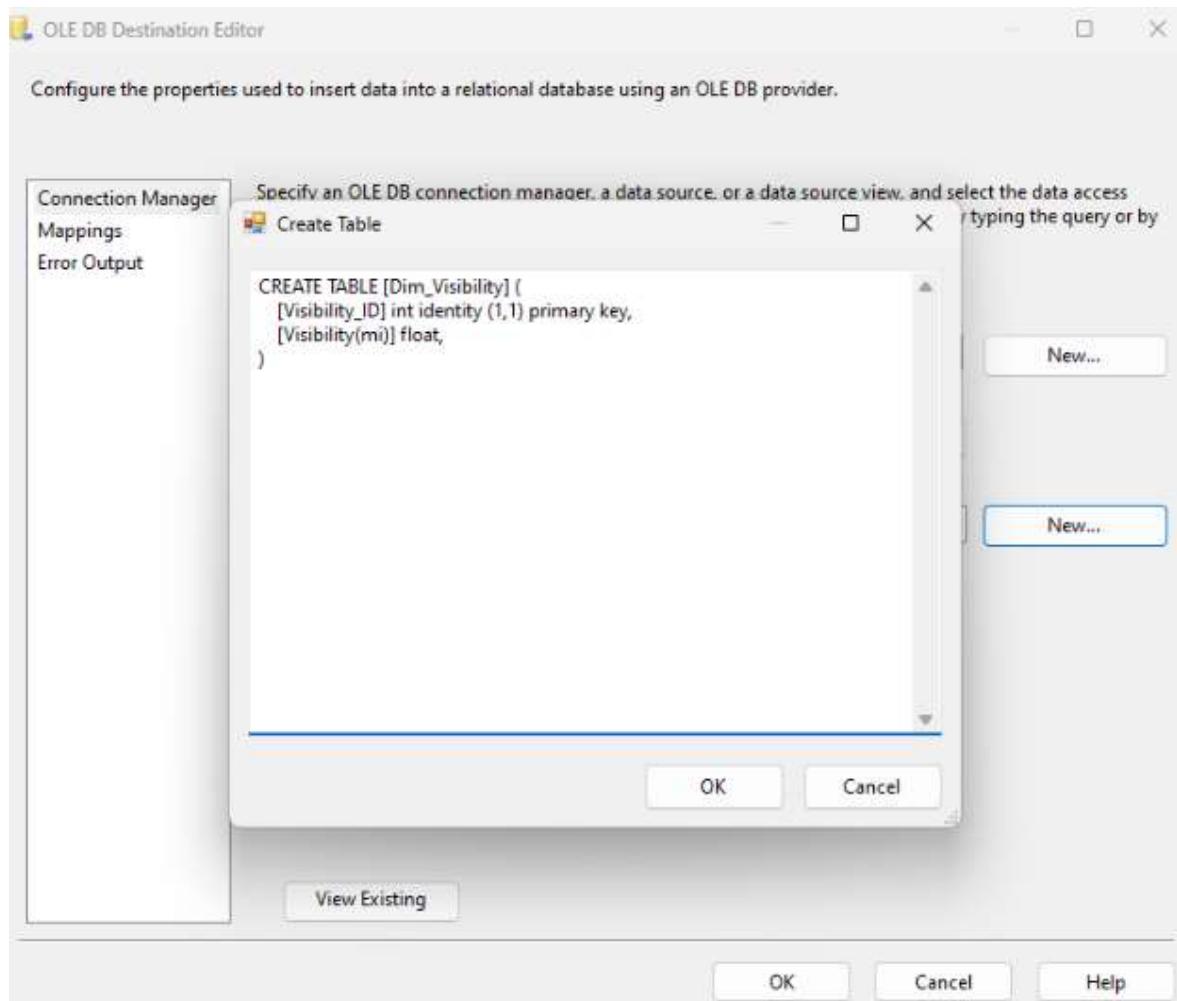


- **Bước 3.** Tạo mới một OLE DB Destination để đổ dữ liệu gốc sau khi đã được xử lý vào trong bảng Dim\_Visibility kho dữ liệu ACCIDENTS DW.



- **Bước 4.** Chọn New... để tạo bảng Dim\_Visibility

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

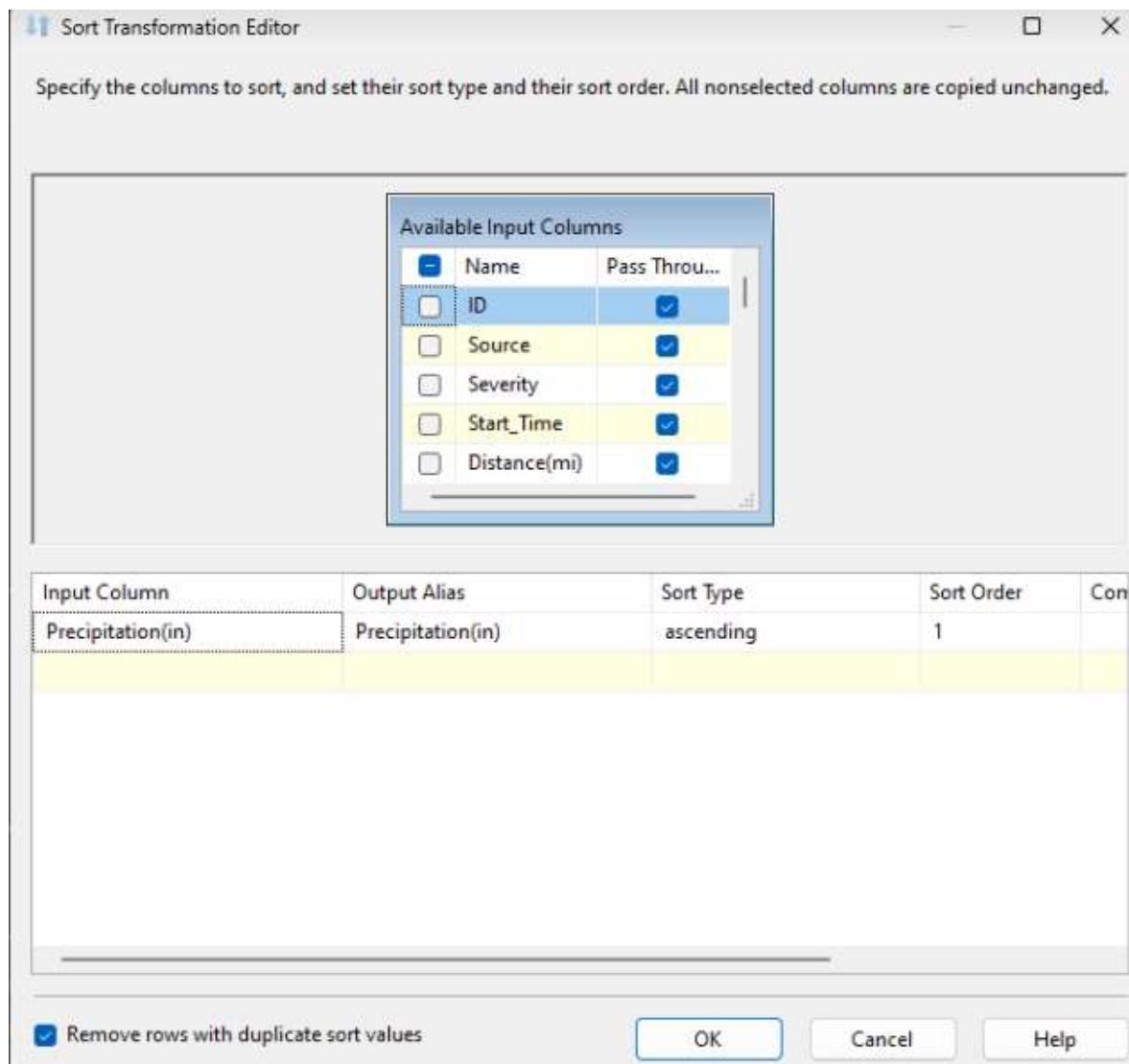


- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu
  - Chọn OK để hoàn tất thiết lập.

### 3.8. Bảng DIM\_PRECIPITATION

- **Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Precipitation cho Dim\_Precipitation.
  - **Bước 2.** Click chuột phải vào Sort\_Dim\_Precipitation, chọn Edit: lần lượt chọn các cột Precipitation làm cột dữ liệu để đổ dữ liệu vào Sort\_Dim\_Precipitation.
    - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

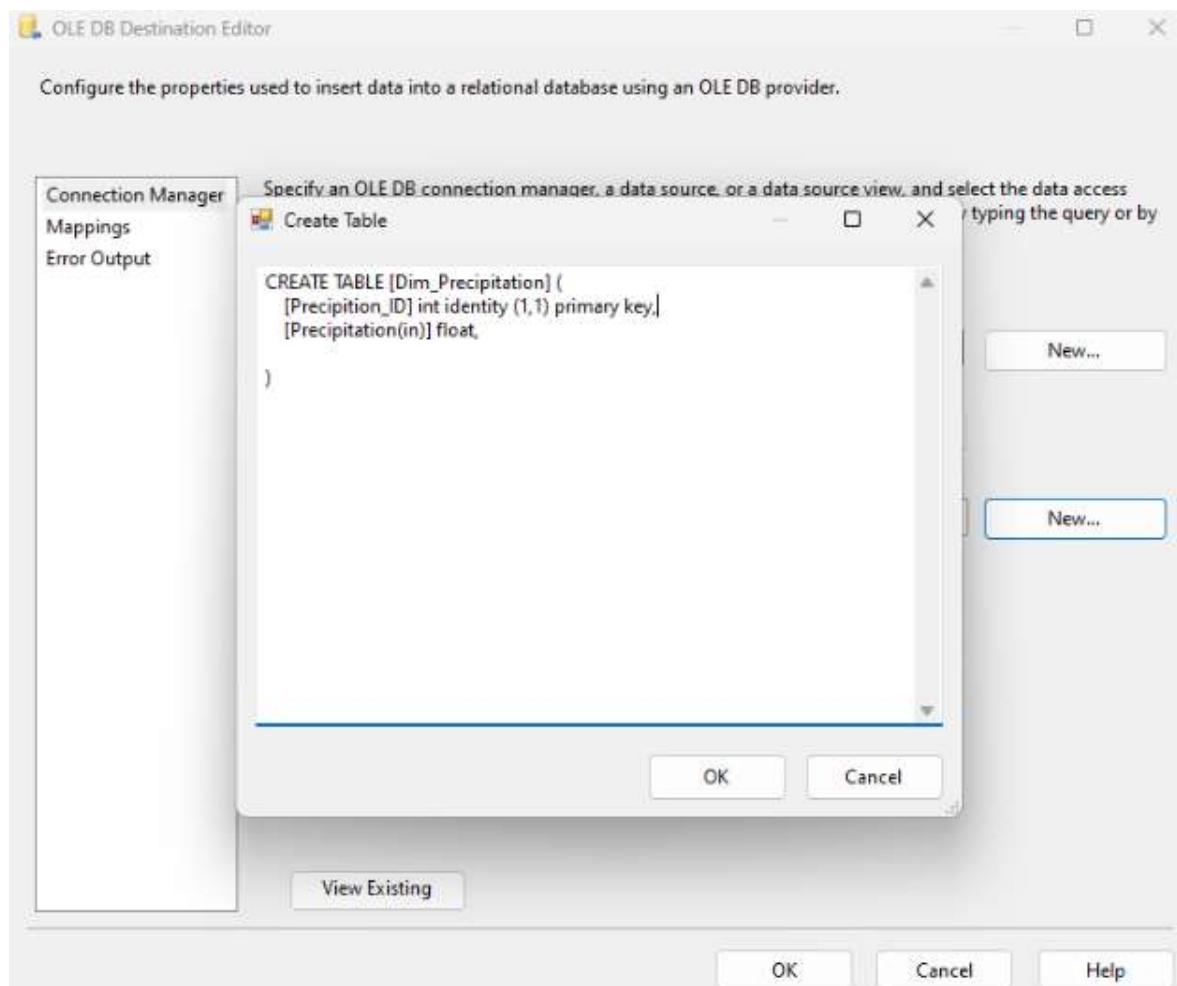


- **Bước 3.** Tạo mới một OLE DB Destination để đổ dữ liệu gốc sau khi đã được xử lý vào trong bảng Dim\_Precipitation của kho dữ liệu ACCIDENTS\_DW.



- **Bước 4.** Chọn New... để tạo bảng Dim\_Precipitation

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



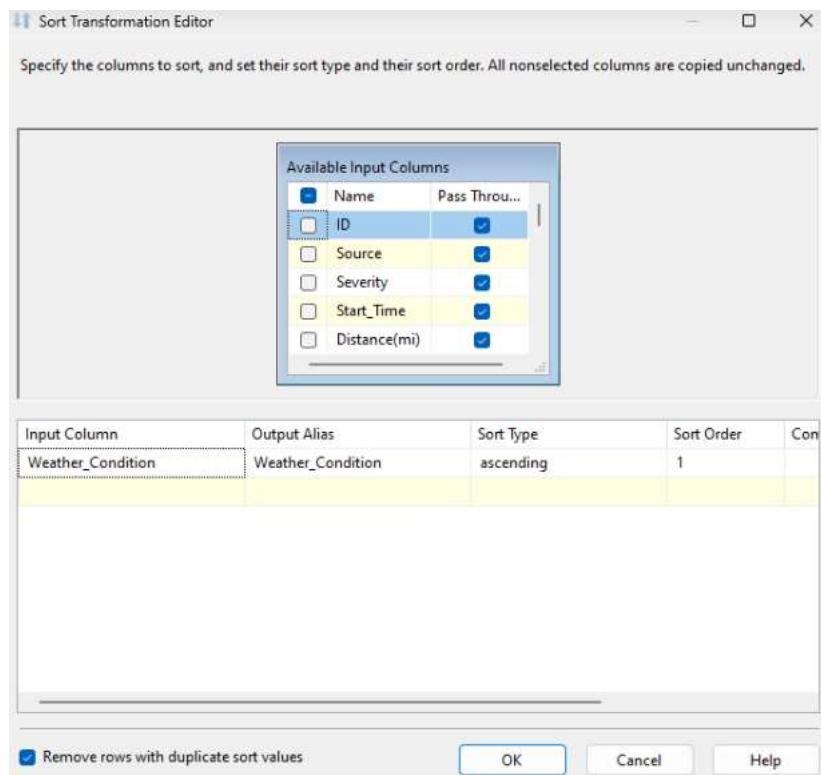
- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

- Chọn OK để hoàn tất thiết lập.

### 3.9. Bảng DIM\_WEATHER

- **Bước 1.** Chọn một Sort để tạo ra Sort\_Dim\_Weather cho Dim\_Weather.
- **Bước 2.** Click chuột phải vào Sort\_Dim\_Weather, chọn Edit: lần lượt chọn các cột Weather\_Condition làm cột dữ liệu để đổ dữ liệu vào Sort\_Dim\_Weather.
  - Tick chọn Remove rows with duplicate sort values xóa đi các dòng dữ liệu trùng nhau và sau đó chọn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

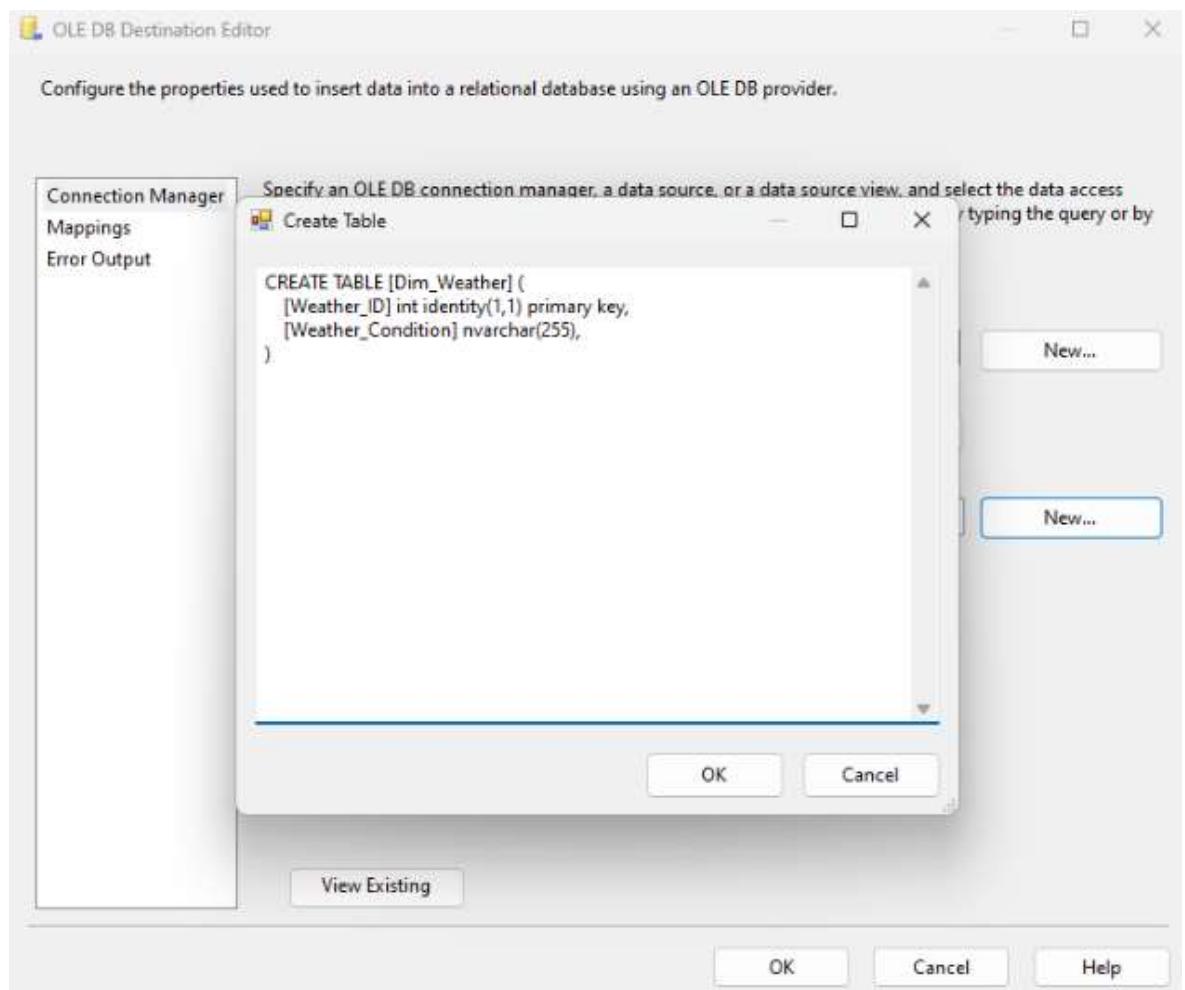


- **Bước 3.** Tạo mới một OLE DB Destination để đỗ dữ liệu gốc sau khi đã được xử lý vào trong bảng Dim\_Weather kho dữ liệu ACCIDENTS\_DW.



- **Bước 4.** Chọn New... để tạo bảng Dim\_Weather

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 5.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

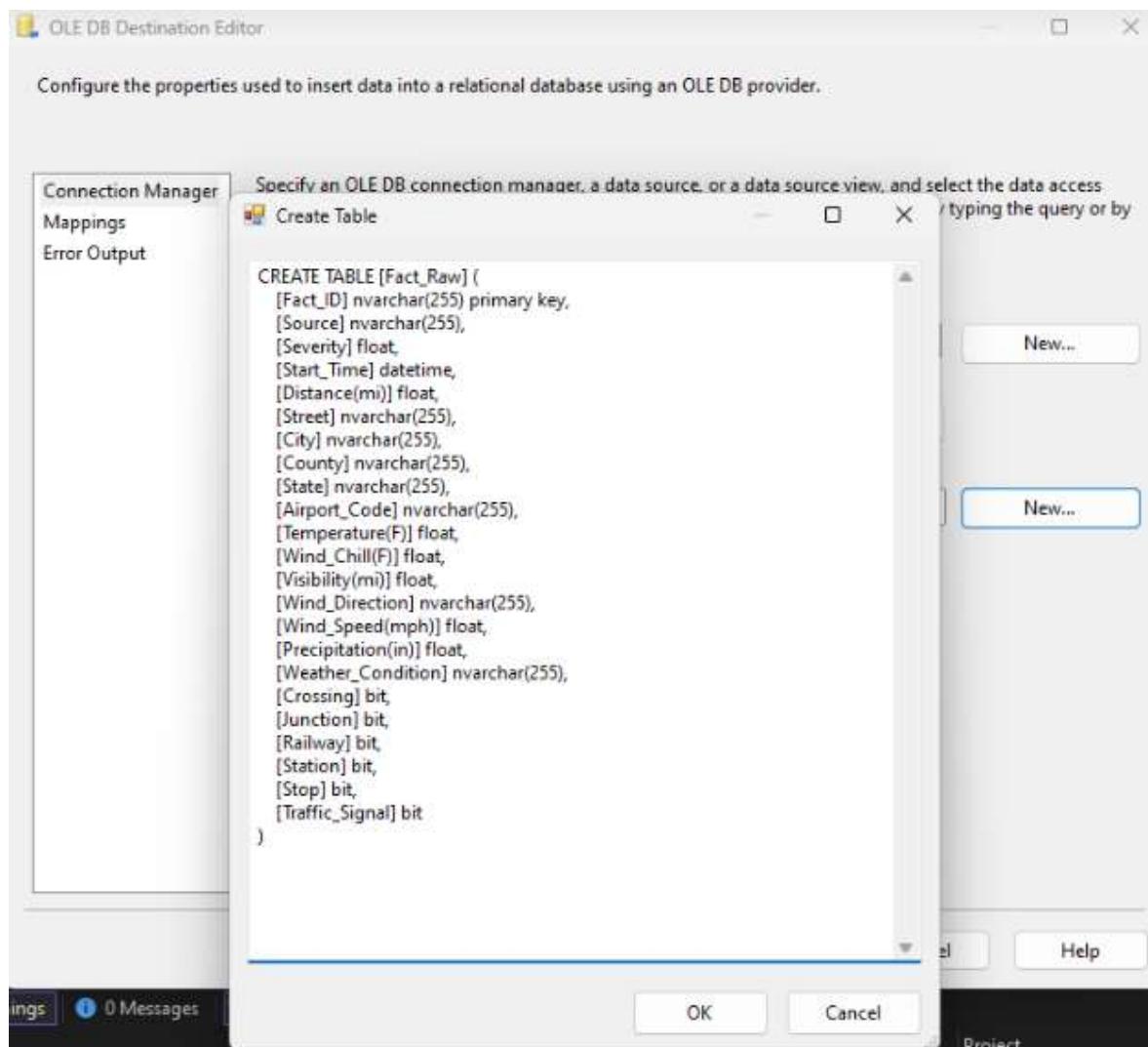
- Chọn OK để hoàn tất thiết lập.

### 3.10. Bảng FACT

- **Bước 1.** Tiến hành tạo bảng Fact và đặt tên là Fact\_Raw từ một OLE DB Destination

- **Bước 2.** Click chuột phải và chọn Edit để tạo bảng Fact\_Raw có các cột là tất cả các cột từ dữ liệu gốc và chứa tất cả các dòng dữ liệu.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



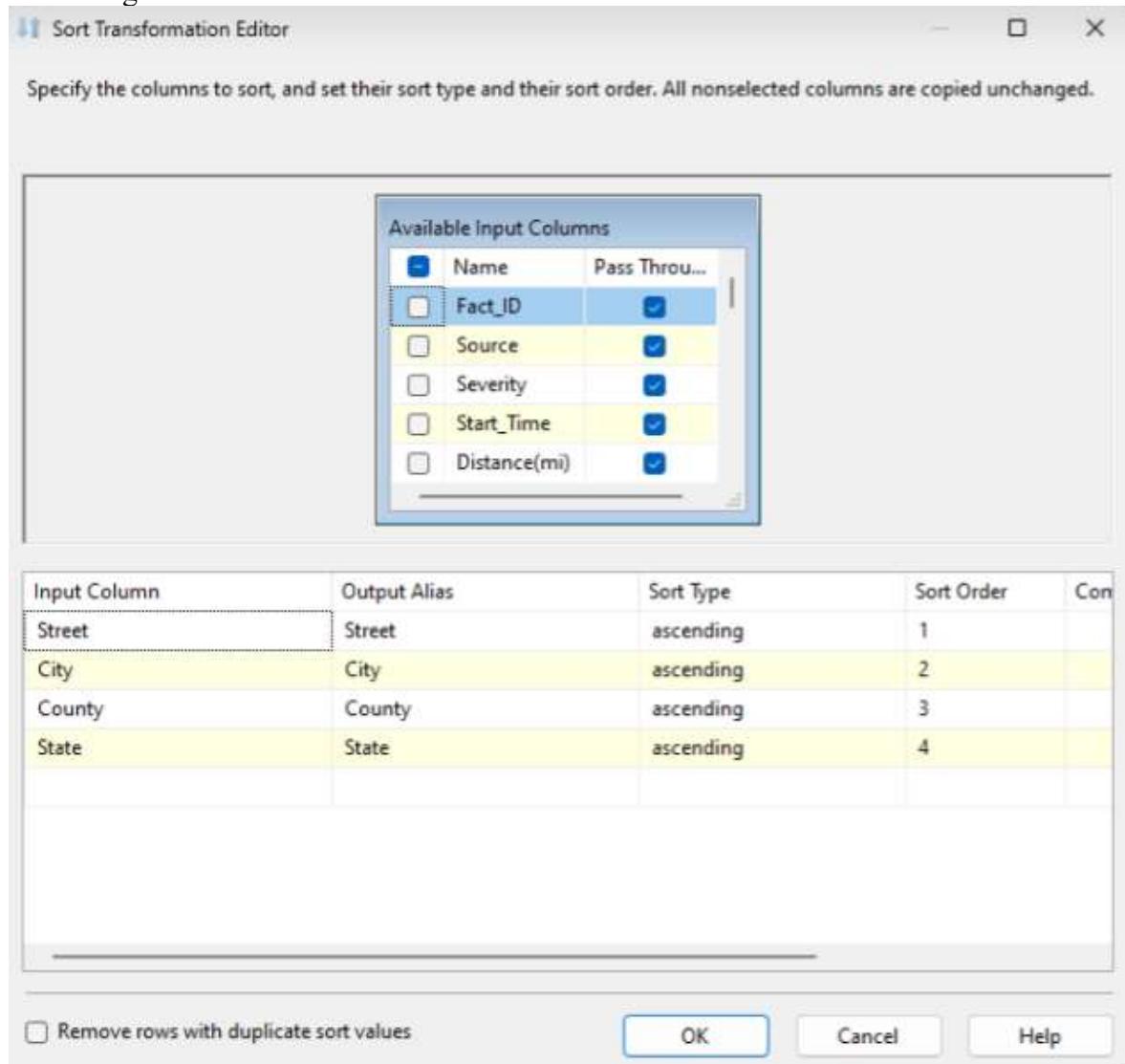
- **Bước 3.** Tiếp đến ta cần chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu. Ta thấy ID của bảng Fact\_Raw cũng chính là ID của tập dữ liệu nên ta chọn ánh xạ cột ID trong Input Column vào cột Fact\_ID của bảng Fact\_Raw
  - Cuối cùng nhấn nút OK để hoàn tất quá trình tạo bảng.
  - Tiếp theo đây ta sẽ thực hiện quá trình lần lượt loại bỏ các cột dữ liệu trùng của bảng Fact với các Dimension, thực hiện thêm khóa ngoại vào bảng Fact nhằm thu gọn bảng Fact, tối ưu hóa quá trình phân tích dữ liệu.

### 3.10.1. Merge Fact\_Raw và Dim\_Location vào Fact1

- **Bước 1.** Ở tab Control Flow, tạo hai Data Flow Task và đổi tên Data Flow Task thứ 2 là “Merge Fact\_Raw and Dim\_Location to Fact1”
- **Bước 2.** Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact\_Raw và Dim\_Location
- **Bước 3.** Click chuột phải chọn Edit, sau đó chọn bảng Fact\_Raw đã tạo trước đó làm data source cho bảng Fact\_Raw mới này.
- **Bước 4.** Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- **Bước 5.** Tương tự thực hiện chọn ánh xạ cột cho Dim\_Location
- **Bước 6.** Tạo 2 Sort là Sort và Sort1 tương ứng với mỗi Source.
- **Bước 7.** Ở Sort, click chuột phải chọn Edit và chọn các cột Street, City, County, State theo thứ tự giống với bảng Dim\_Location để chuẩn bị cho quá

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

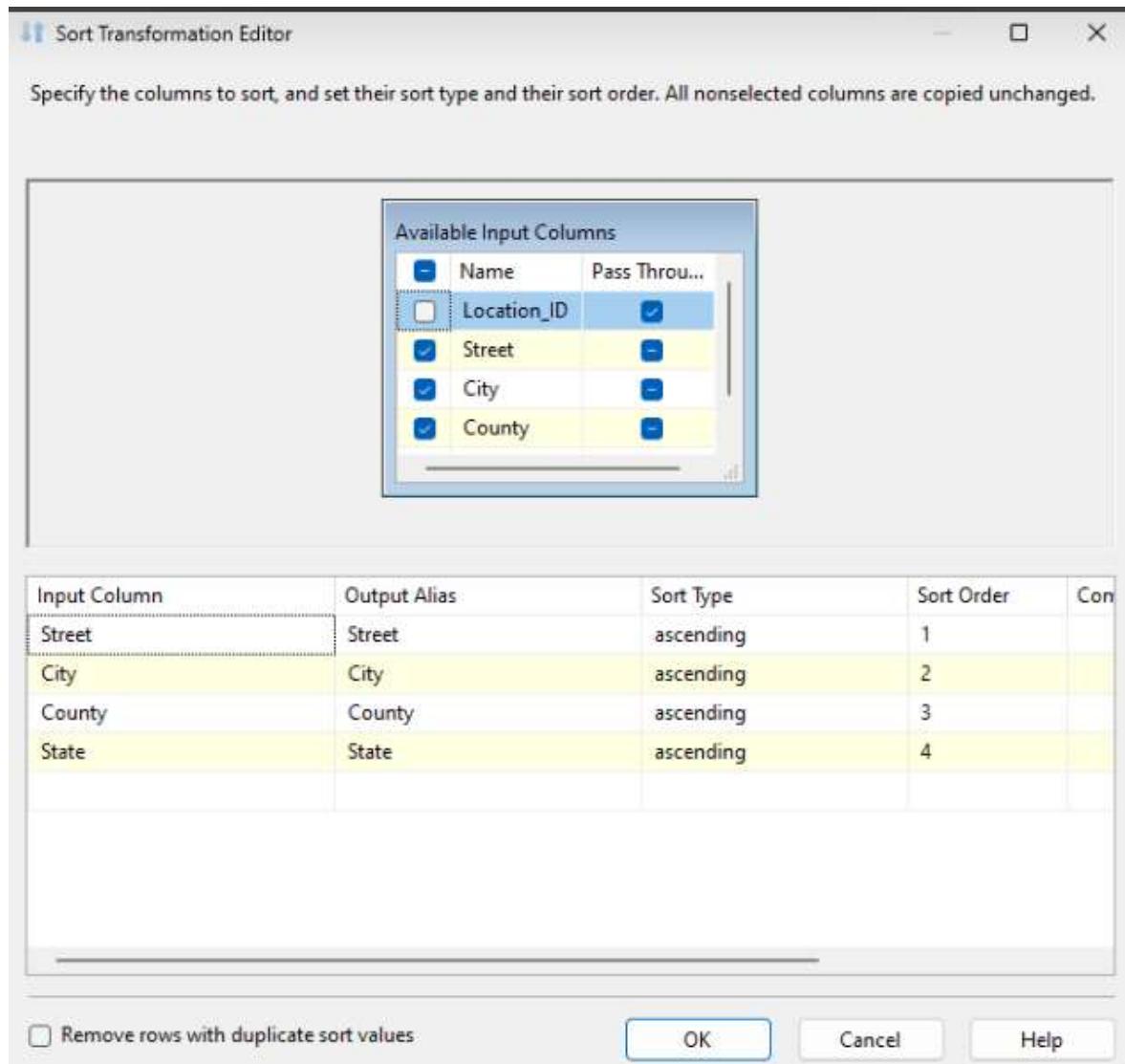
trình merge.



- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo ta chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact\_Raw bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_Location hay không.

- **Bước 9.** Tương tự ta chọn các cột Street, City, County và State cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Nối Sort1 với Merge Join
- **Bước 10.**
  - Chuột phải vào Merge Join và nhấn Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy 4 thuộc tính Street, City, County và State.
    - Tiếp theo ta chọn Location\_ID ở Sort1 để merge vào Fact\_Raw
    - Kết quả sau khi merge là bảng Fact\_Raw không còn 4 thuộc tính Street, City, County và State và có thêm 1 thuộc tính mới là Location\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Join type: Inner join Swap Inputs

Sort

Name	Order	Join K...
Fact_ID	0	<input type="checkbox"/>
Source	0	<input type="checkbox"/>
Severity	0	<input type="checkbox"/>
Start_Time	0	<input type="checkbox"/>
Distance(mi)	0	<input type="checkbox"/>
Street	1	<input checked="" type="checkbox"/>
City	2	<input checked="" type="checkbox"/>
County	3	<input checked="" type="checkbox"/>
State	4	<input checked="" type="checkbox"/>
Airport_Co...	0	<input type="checkbox"/>
Temperatu...	0	<input type="checkbox"/>
Wind_Chill...	0	<input type="checkbox"/>
Visibility(mi)	0	<input type="checkbox"/>
Wind_Dire...	0	<input type="checkbox"/>
Wind_Spee...	0	<input type="checkbox"/>
Precipitati...	0	<input type="checkbox"/>
Weather_C...	0	<input type="checkbox"/>
Crossing	0	<input type="checkbox"/>
Junction	0	<input type="checkbox"/>
Railway	0	<input type="checkbox"/>
Station	0	<input type="checkbox"/>
Stop	0	<input type="checkbox"/>
Traffic_Sig...	0	<input type="checkbox"/>

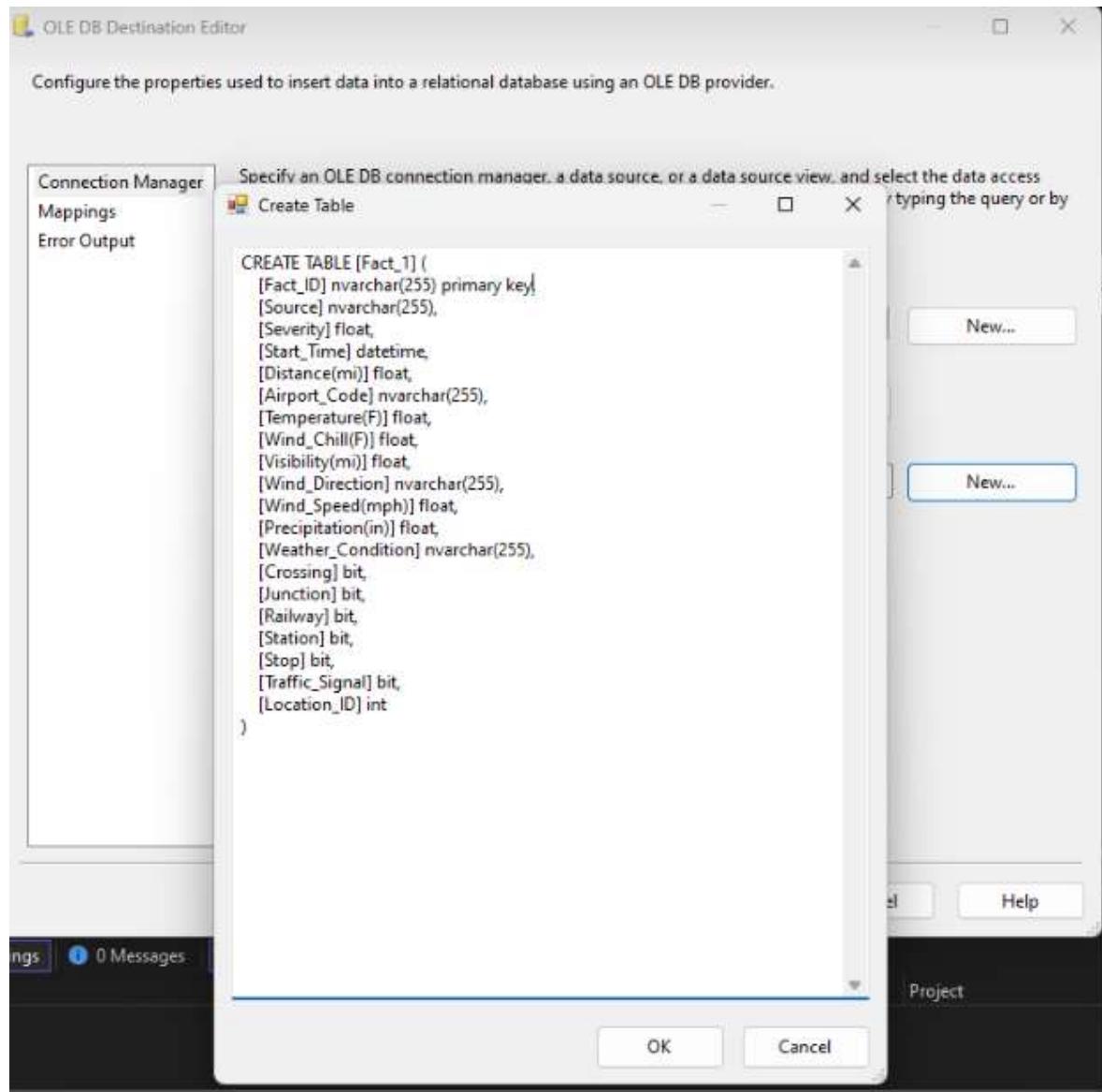
Sort 1

Name	Order	Join K...
Location_ID	0	<input type="checkbox"/>
Street	1	<input checked="" type="checkbox"/>
City	2	<input checked="" type="checkbox"/>
County	3	<input checked="" type="checkbox"/>
State	4	<input checked="" type="checkbox"/>

Input	Input Column	Output Alias
Sort	Fact_ID	Fact_ID
Sort	Source	Source
Sort	Severity	Severity
Sort	Start_Time	Start_Time
Sort	Distance(mi)	Distance(mi)
Sort	Airport_Code	Airport_Code
Sort	Temperature(F)	Temperature(F)

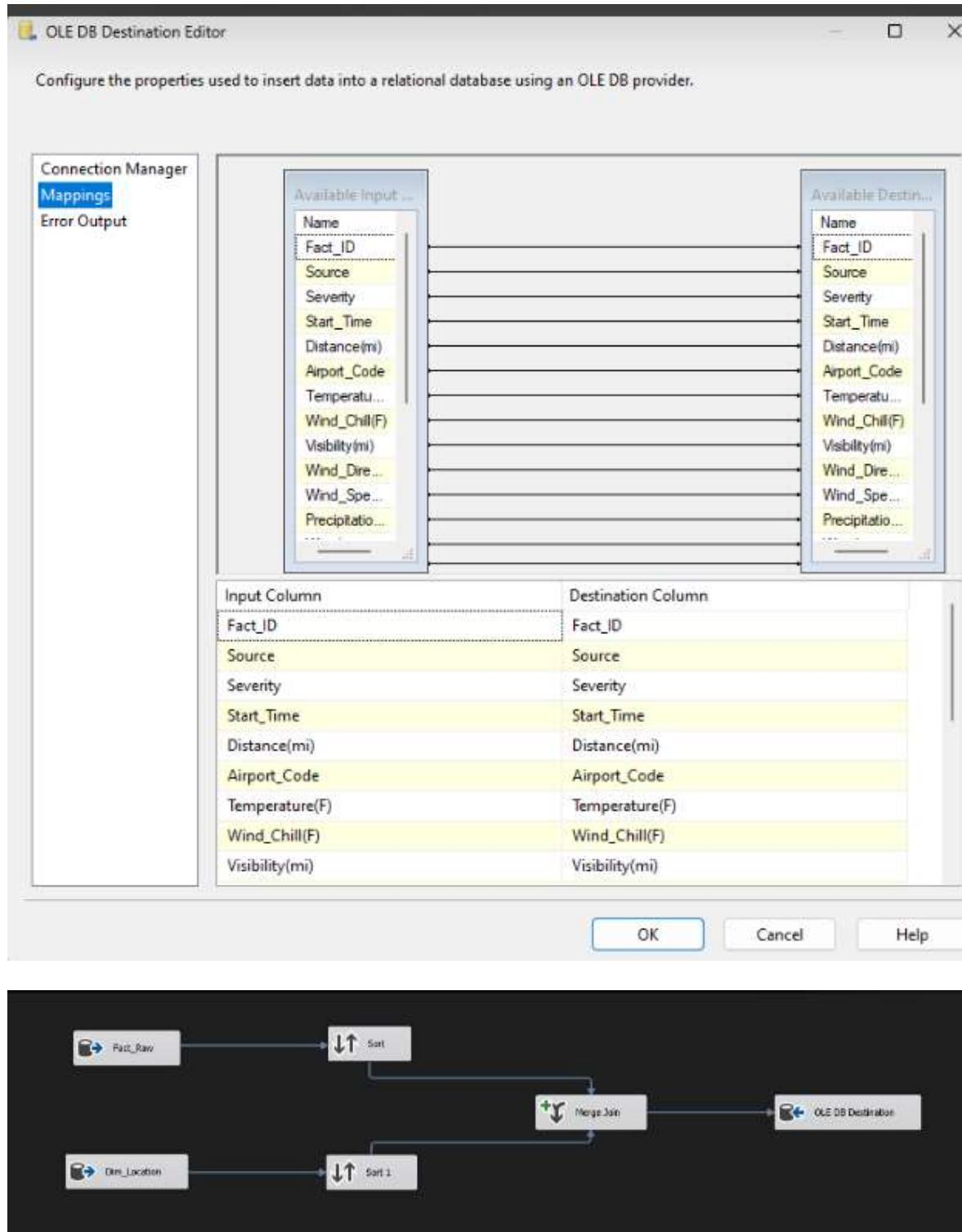
- **Bước 11.** Tạo bảng Fact1 từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



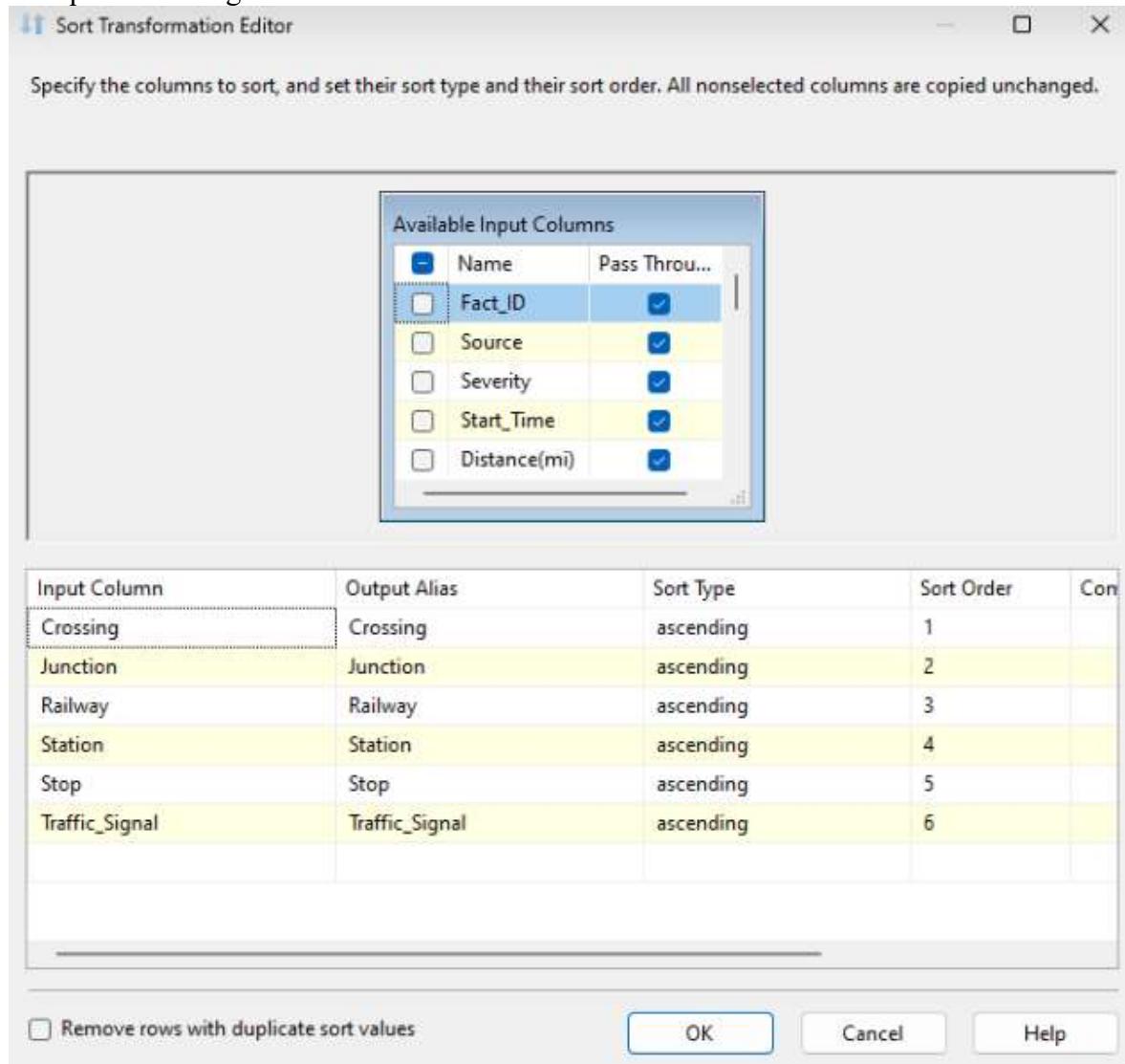
- Tiếp tục quá trình merge bảng Fact1 với các Dimension còn lại để có 1 bảng Fact hoàn chỉnh.

### 3.10.2. Merge Fact1 và Dim\_Traffic\_Condition vào Fact2

- **Bước 1.** Ở tab Control Flow, tạo thêm một Data Flow Task và đổi tên Data Flow Task này là “Merge Fact1 and Dim\_Traffic\_Condition to Fact2”
- **Bước 2.** Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact1 và Dim\_Traffic\_Condition

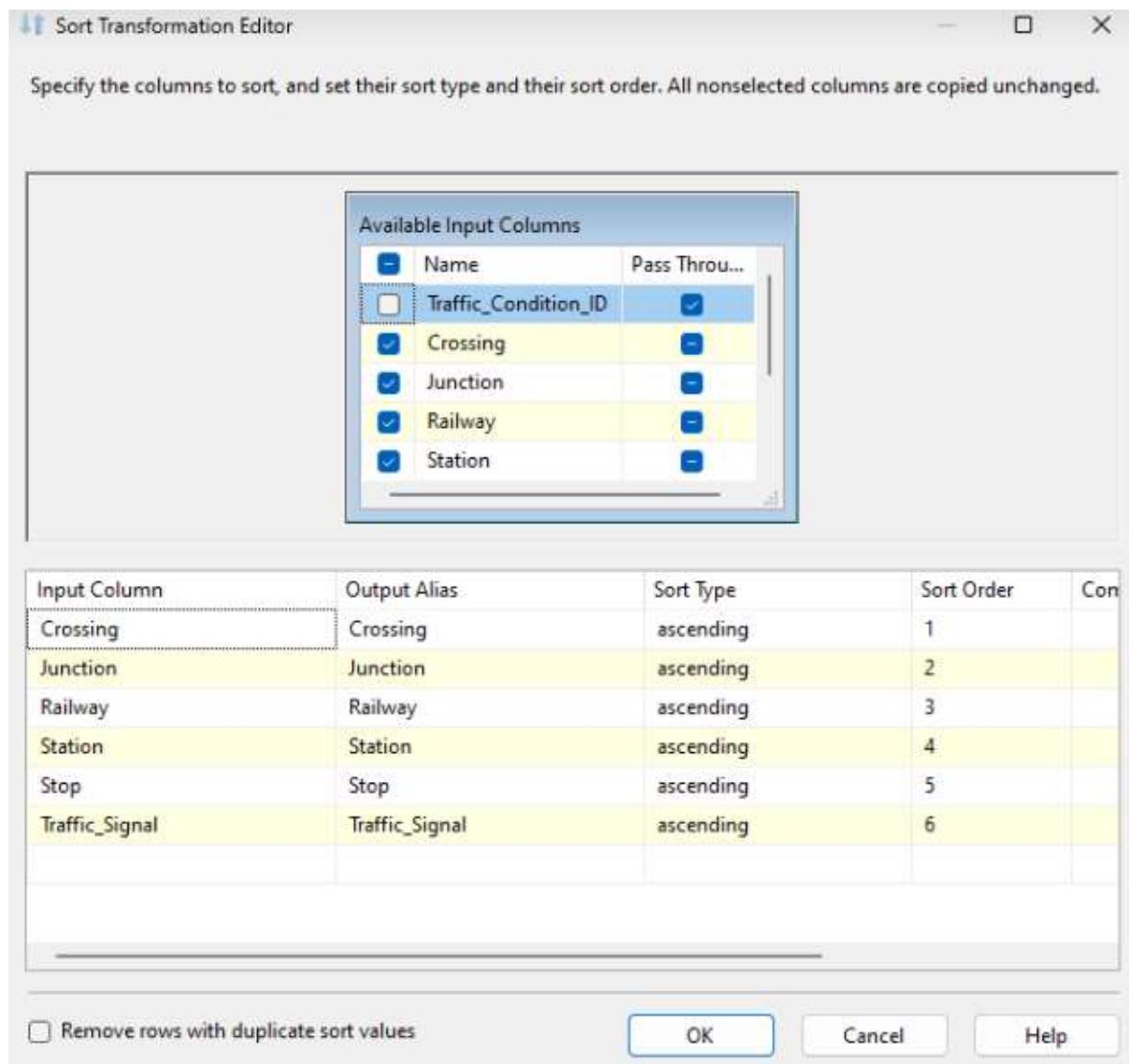
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- **Bước 3.** Click chuột phải vào Fact1 chọn Edit, sau đó chọn bảng Fact1 đã được tạo khi merge Fact\_Raw và Dim\_Location làm data source.
- **Bước 4.** Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- **Bước 5.** Thực hiện chọn ánh xạ các cột cho Dim\_Traffic\_Condition
  - Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- **Bước 6.** Tạo 2 Sort tương ứng với mỗi Source
- **Bước 7.** Tại Sort, click chuột phải chọn Edit và chọn các cột Bump, Crossing, Junction, Railway, Roundabout, Station, Stop, Traffic\_Calming và Traffic\_Signal theo thứ tự giống với bảng Dim\_Traffic\_Conditon để chuẩn bị cho quá trình merge



- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact1 bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_Traffic\_Conditon hay không.
- **Bước 9.** Tương tự ta chọn các cột Bump, Crossing, Junction, Railway, Roundabout, Station, Stop, Traffic\_Calming và Traffic\_Signal cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

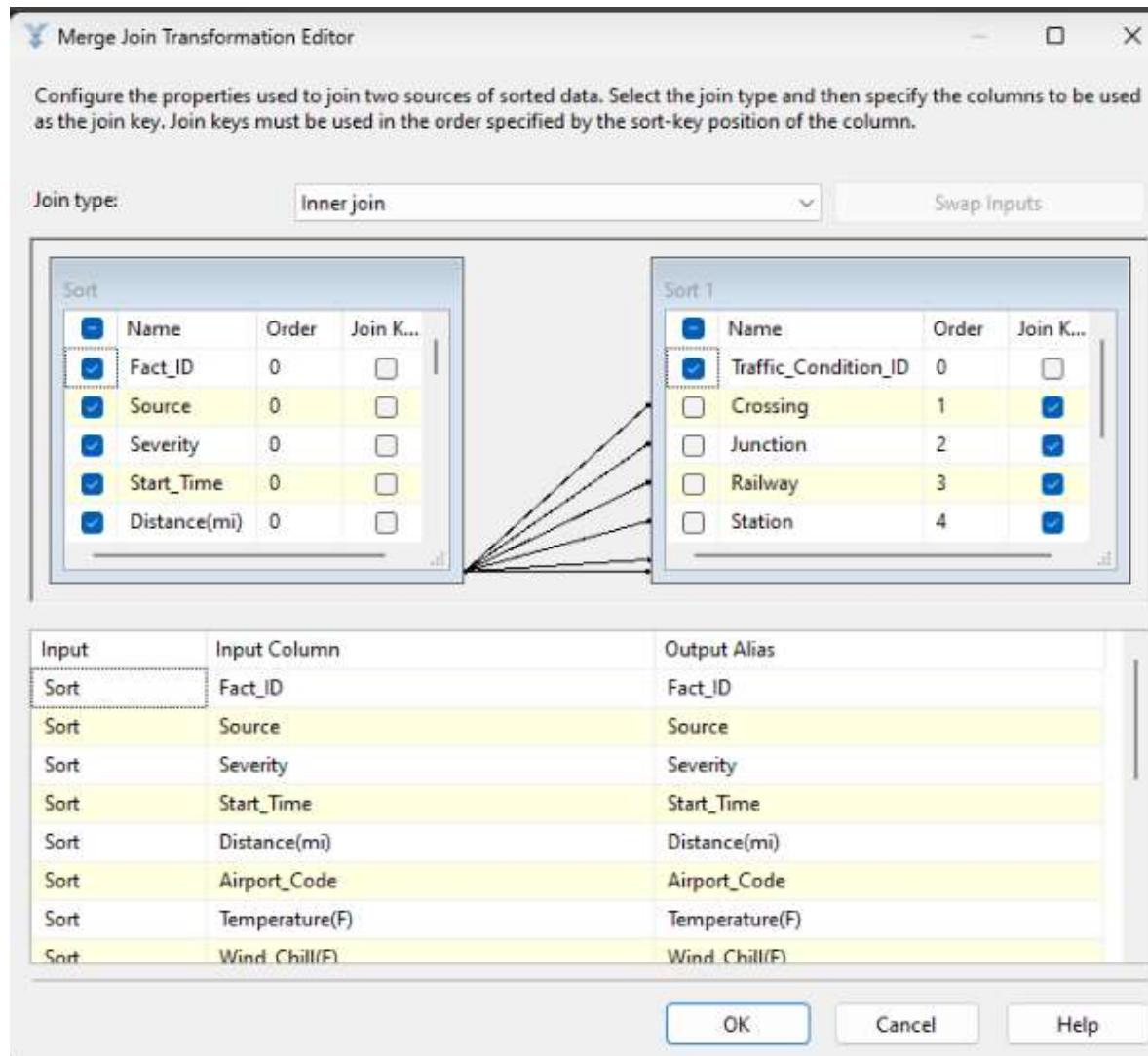


- Nối Sort1 với Merge Join

### - Bước 10.

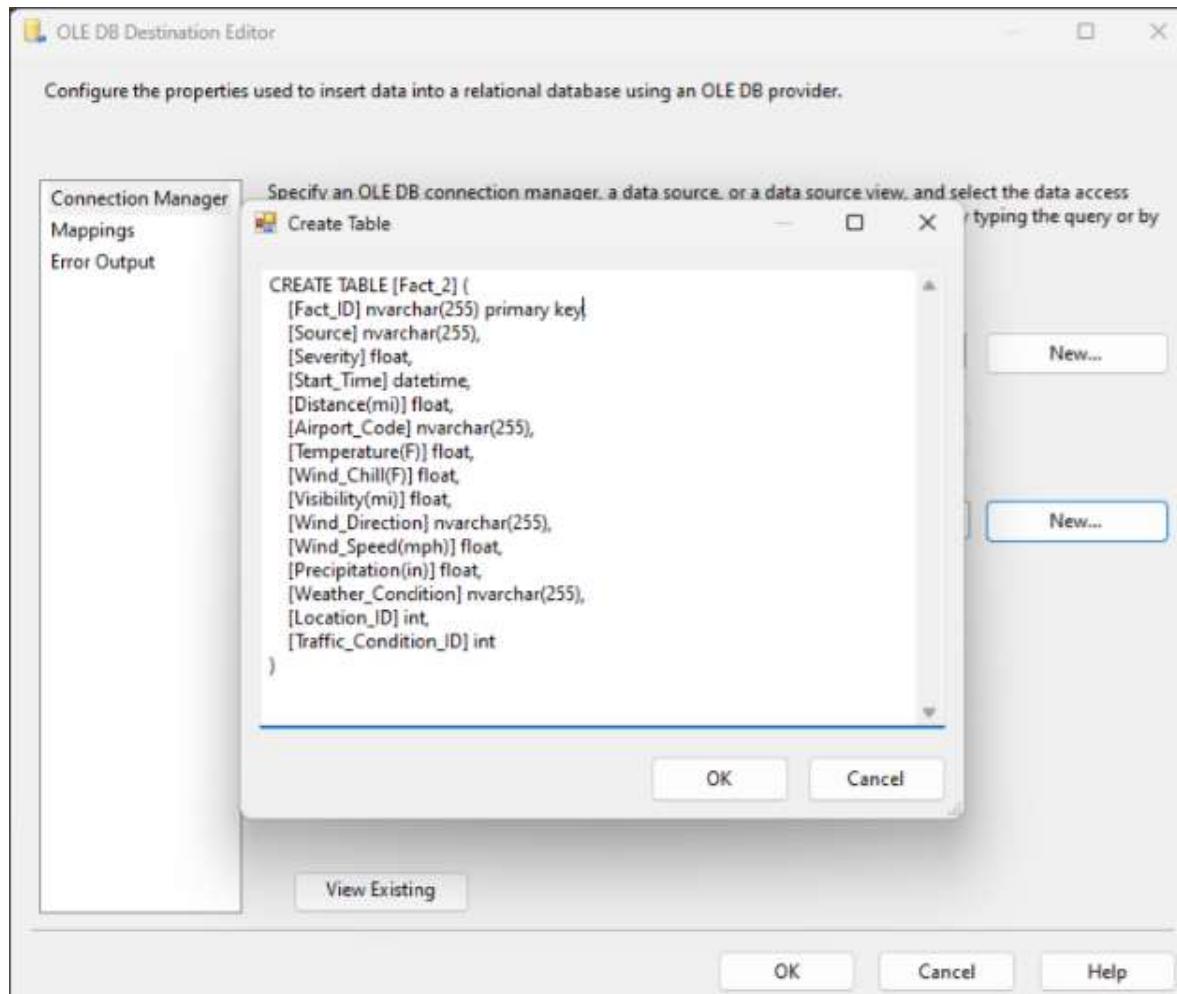
- Chuột phải vào Merge Join và nhấn Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy 9 thuộc tính Bump, Crossing, Junction, Railway, Roundabout, Station, Stop, Traffic\_Calming và Traffic\_Signal.
  - Tiếp theo ta chọn Traffic\_Condition\_ID ở Sort1 để merge vào Fact1
  - Kết quả sau khi merge là bảng Fact1 không còn 9 thuộc tính Bump, Crossing, Junction, Railway, Roundabout, Station, Stop, Traffic\_Calming và Traffic\_Signal và có thêm 1 thuộc tính mới là Traffic\_Condition\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

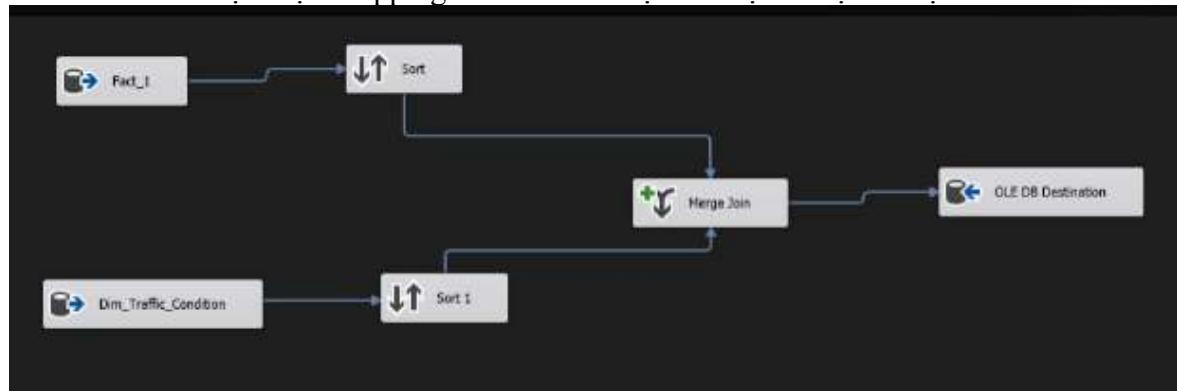


- **Bước 11.** Tạo bảng Fact2 từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

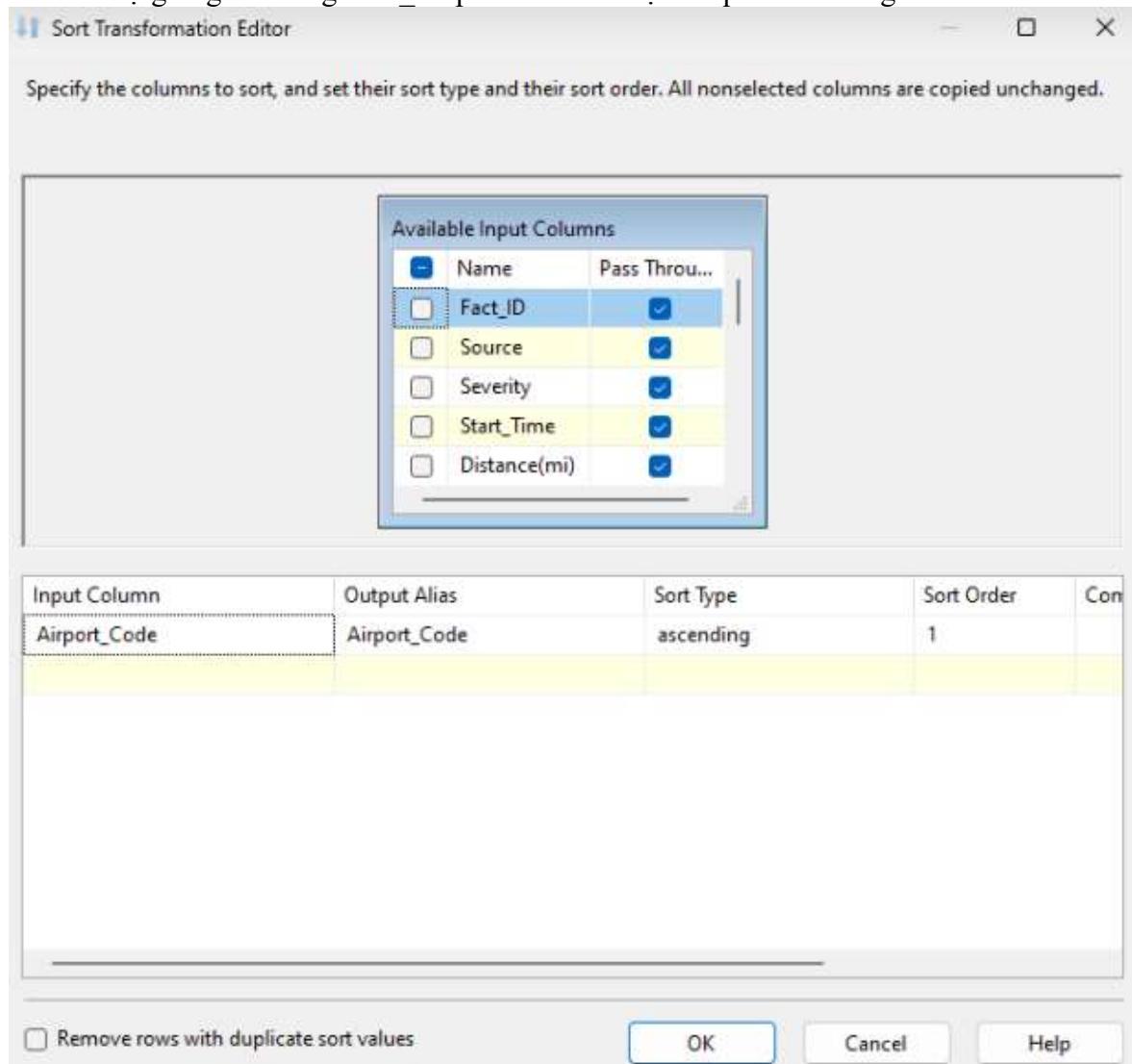


### 3.10.3. Merge Fact2 và Dim\_Airport vào Fact3

- Bước 1.** Ở tab Control Flow, tạo thêm một Data Flow Task và đổi tên Data Flow Task này là “Merge Fact2 and Dim\_Airport to Fact3”
- Bước 2.** Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact2 và Dim\_Airport
- Bước 3.** Click chuột phải vào Fact2 chọn Edit, sau đó chọn bảng Fact2 đã được tạo khi merge Fact1 và Dim\_Traffic\_Condition làm data source.
- Bước 4.** Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- Bước 5.** Thực hiện chọn ánh xạ các cột cho Dim\_Airport

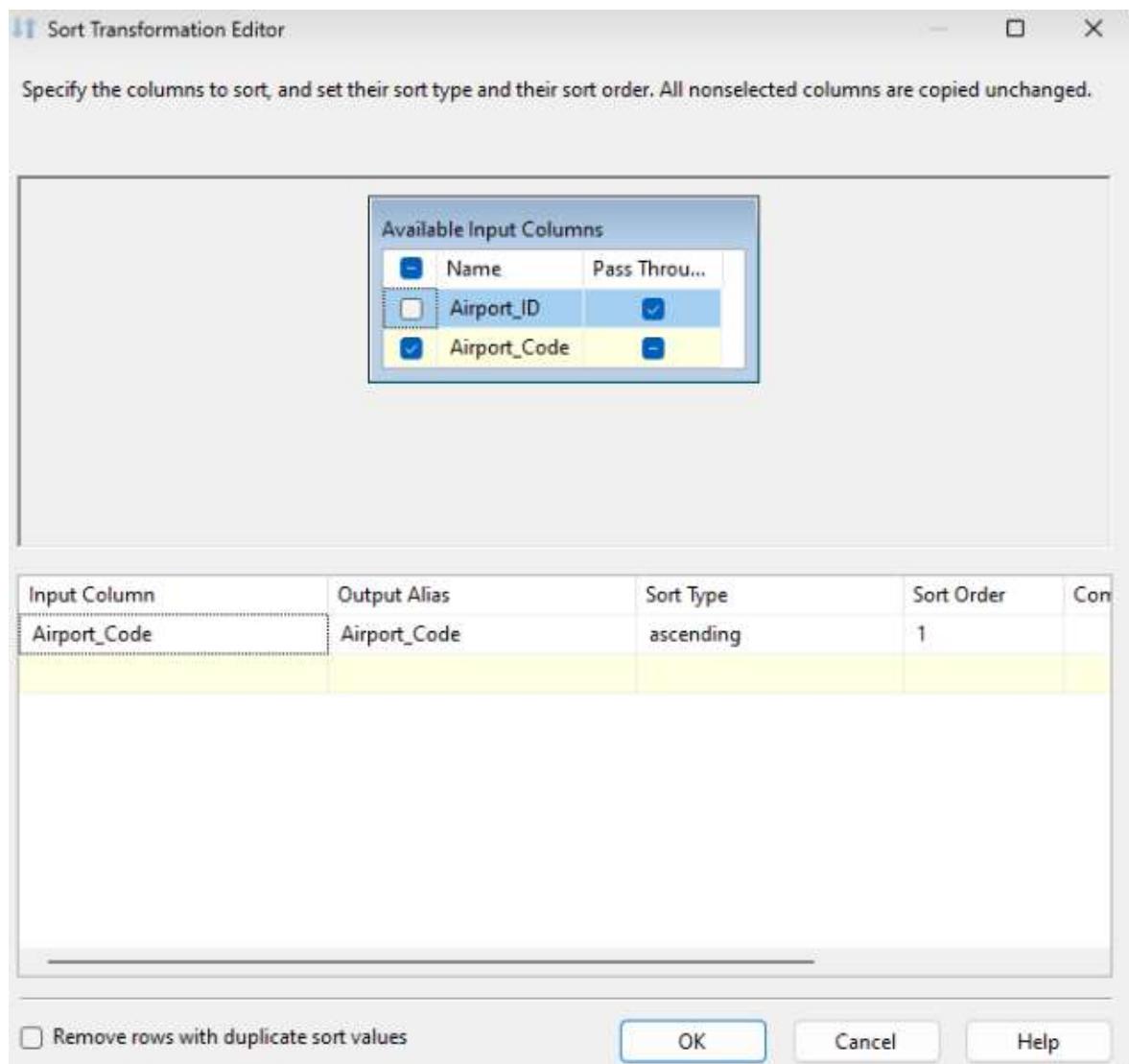
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- **Bước 6.** Tạo 2 Sort tương ứng với mỗi Source
- **Bước 7.** Tại Sort, click chuột phải chọn Edit và chọn cột Airport\_Code theo thứ tự giống với bảng Dim\_Airport để chuẩn bị cho quá trình merge



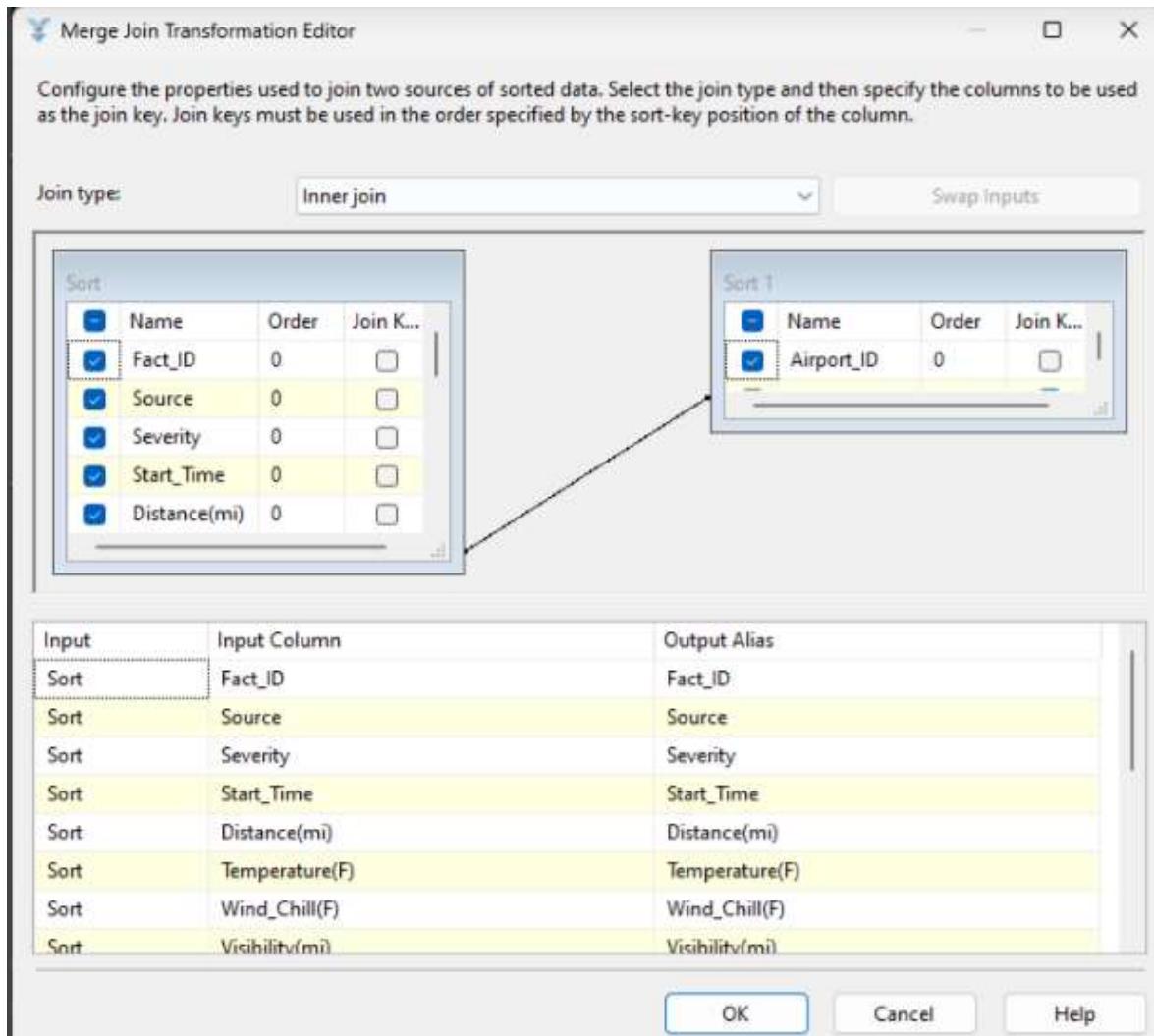
- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact2 bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_Airport hay không.
- **Bước 9.** Tương tự ta chọn cột Airport\_Code cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



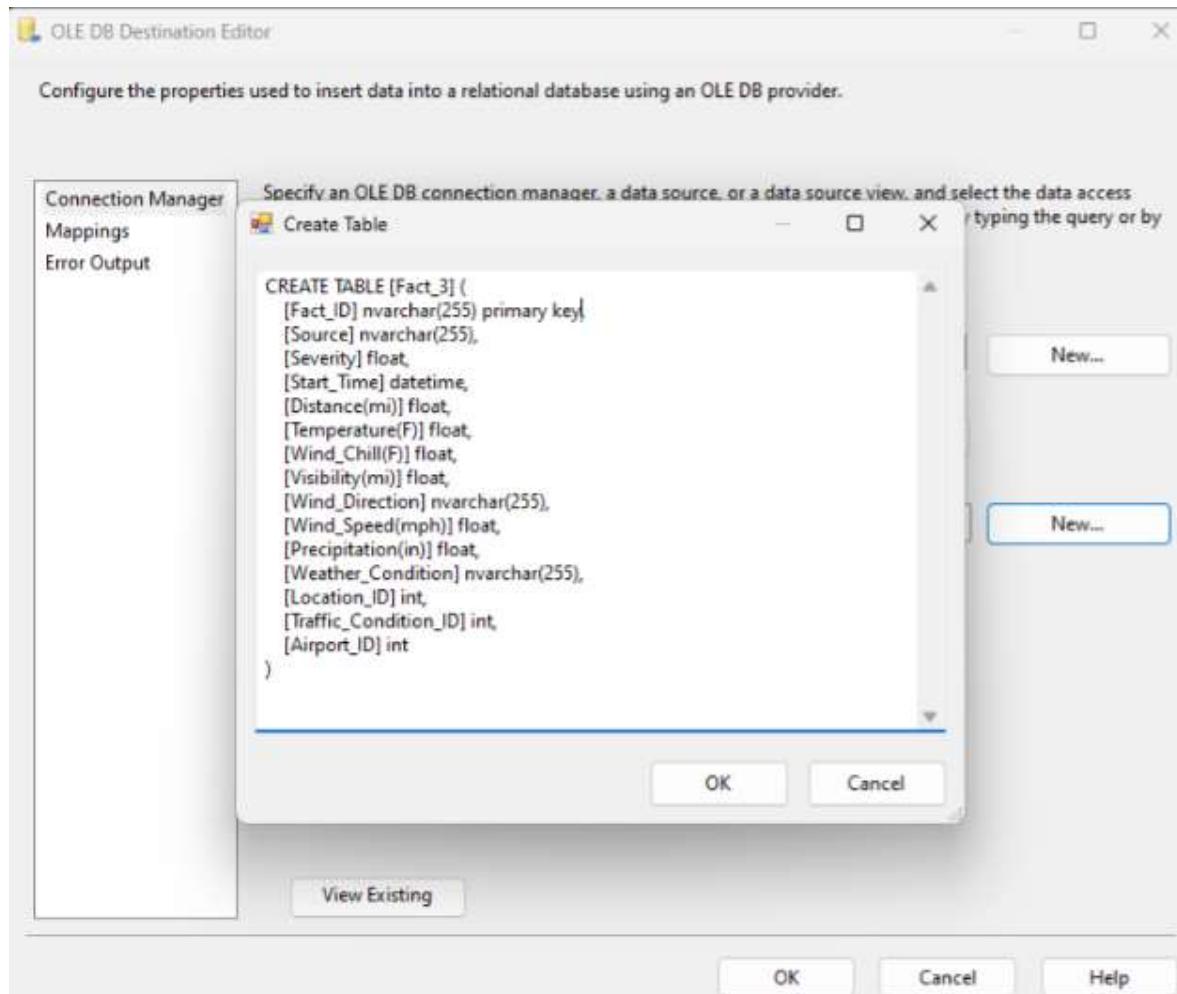
- Nối Sort1 với Merge Join
- **Bước 10.**
  - Chuột phải vào Merge Join và nhấn Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy thuộc tính Airport\_Code.
    - Tiếp theo ta chọn Aiport\_ID ở Sort1 để merge vào Fact2
    - Kết quả sau khi merge là bảng Fact2 không còn thuộc tính Airport\_Code và có thêm 1 thuộc tính mới là Airport\_Record\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

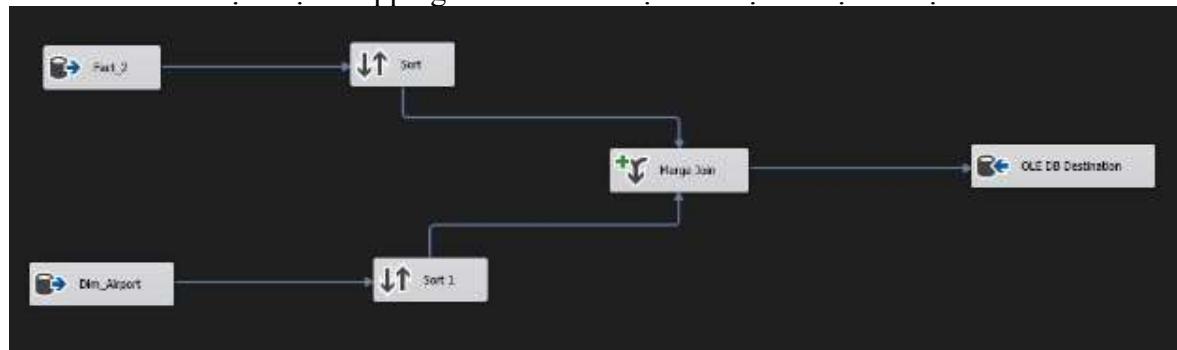


- **Bước 11.** Tạo bảng Fact3 từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu



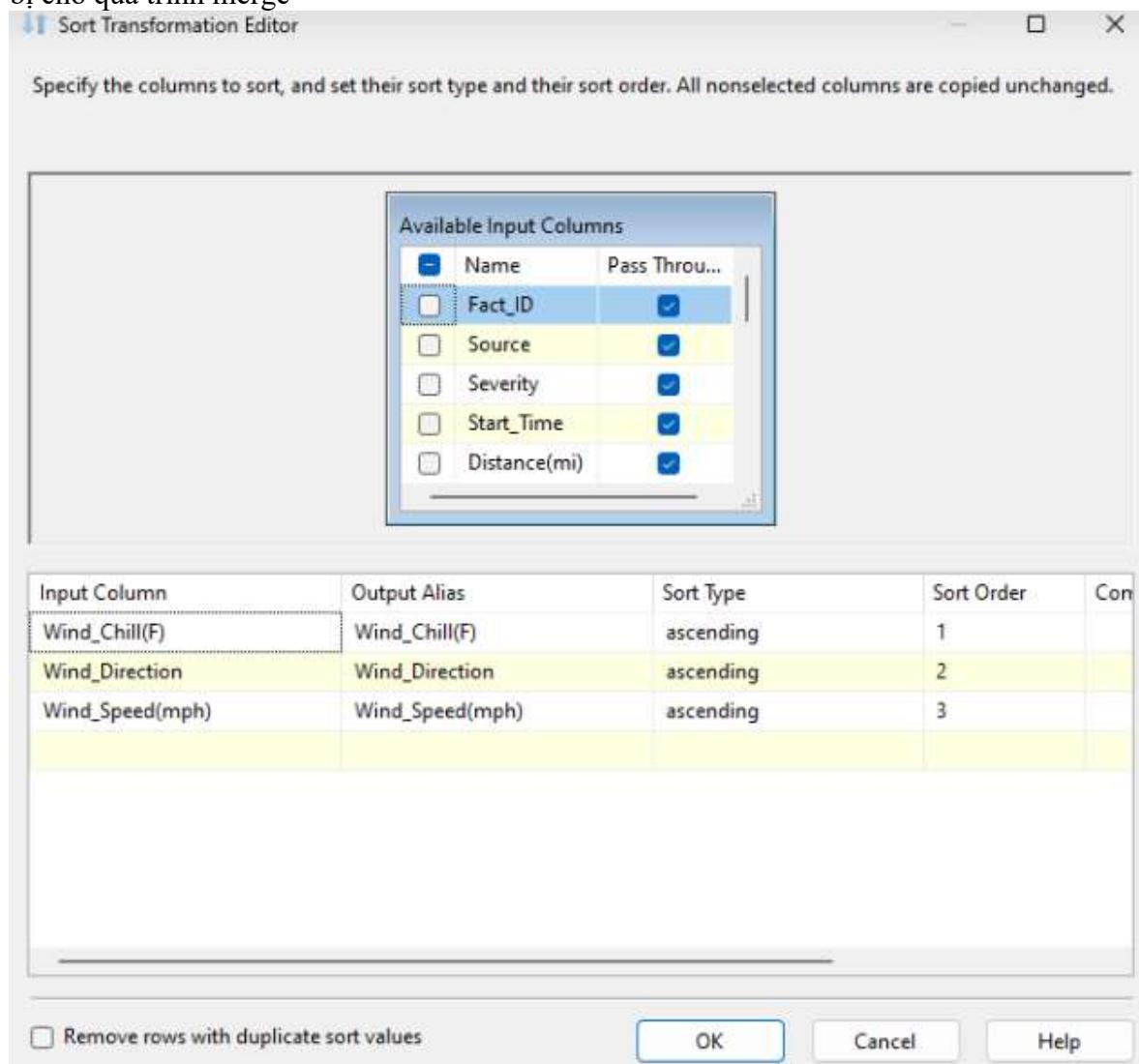
### 3.10.4. Merge Fact3 và Dim\_Wind vào Fact4

- Bước 1.** Ở tab Control Flow, tạo thêm một Data Flow Task và đổi tên Data Flow Task này là “Merge Fact3 and Dim\_Wind to Fact4”
- Bước 2.** Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact3 và Dim\_Wind
  - Bước 3.** Click chuột phải vào Fact3 chọn Edit, sau đó chọn bảng Fact3 đã được tạo khi merge Fact2 và Dim\_Airport làm data source.
  - Bước 4.** Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
  - Bước 5.** Thực hiện chọn ánh xạ các cột cho Dim\_Airport
    - Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- **Bước 6.** Tạo 2 Sort tương ứng với mỗi Source

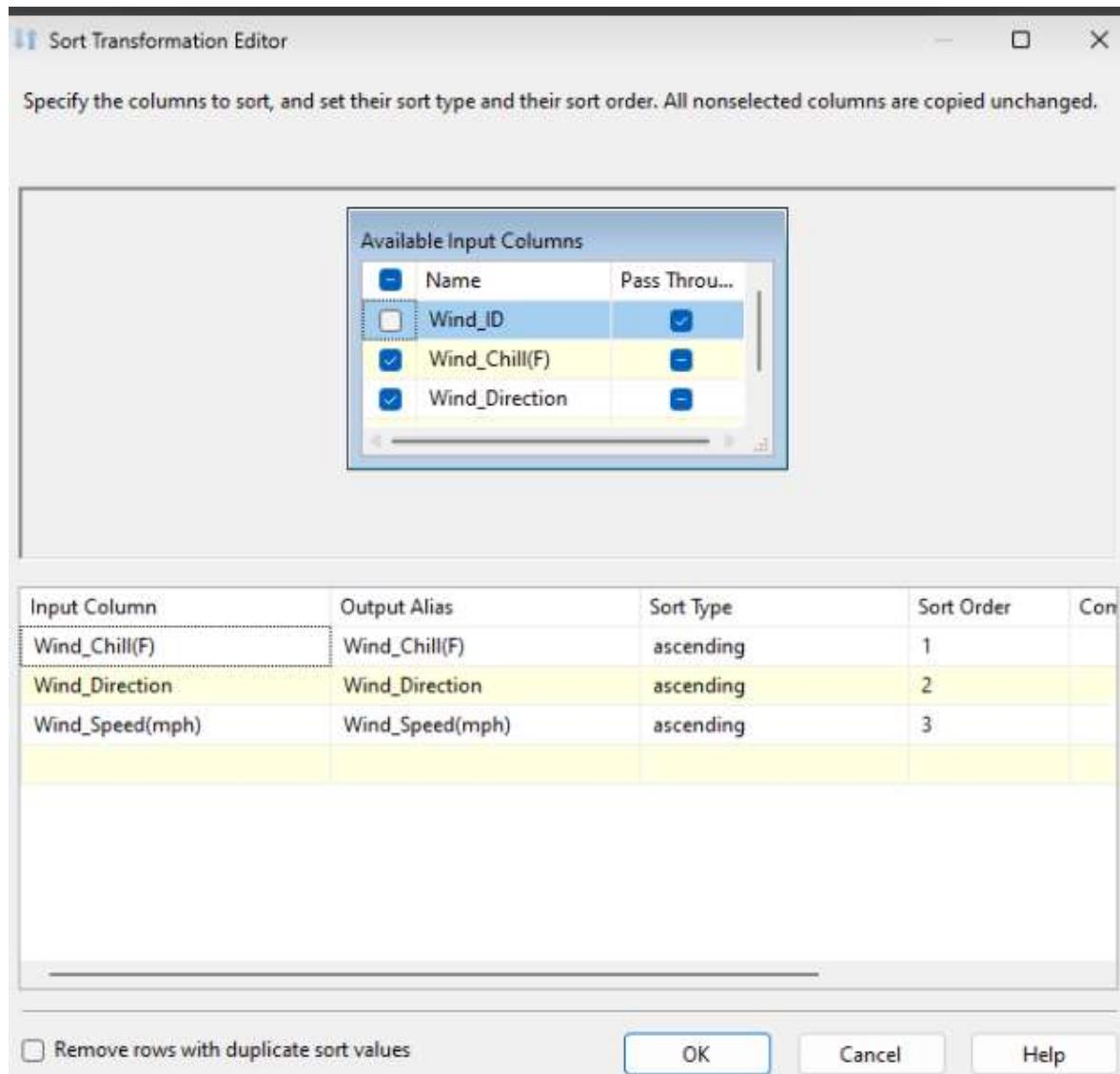
- **Bước 7.** Tại Sort, click chuột phải chọn Edit và chọn 2 cột Wind\_Direction, Wind\_Speed theo thứ tự giống với bảng Dim\_Wind để chuẩn bị cho quá trình merge



- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact3 bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_Wind hay không.

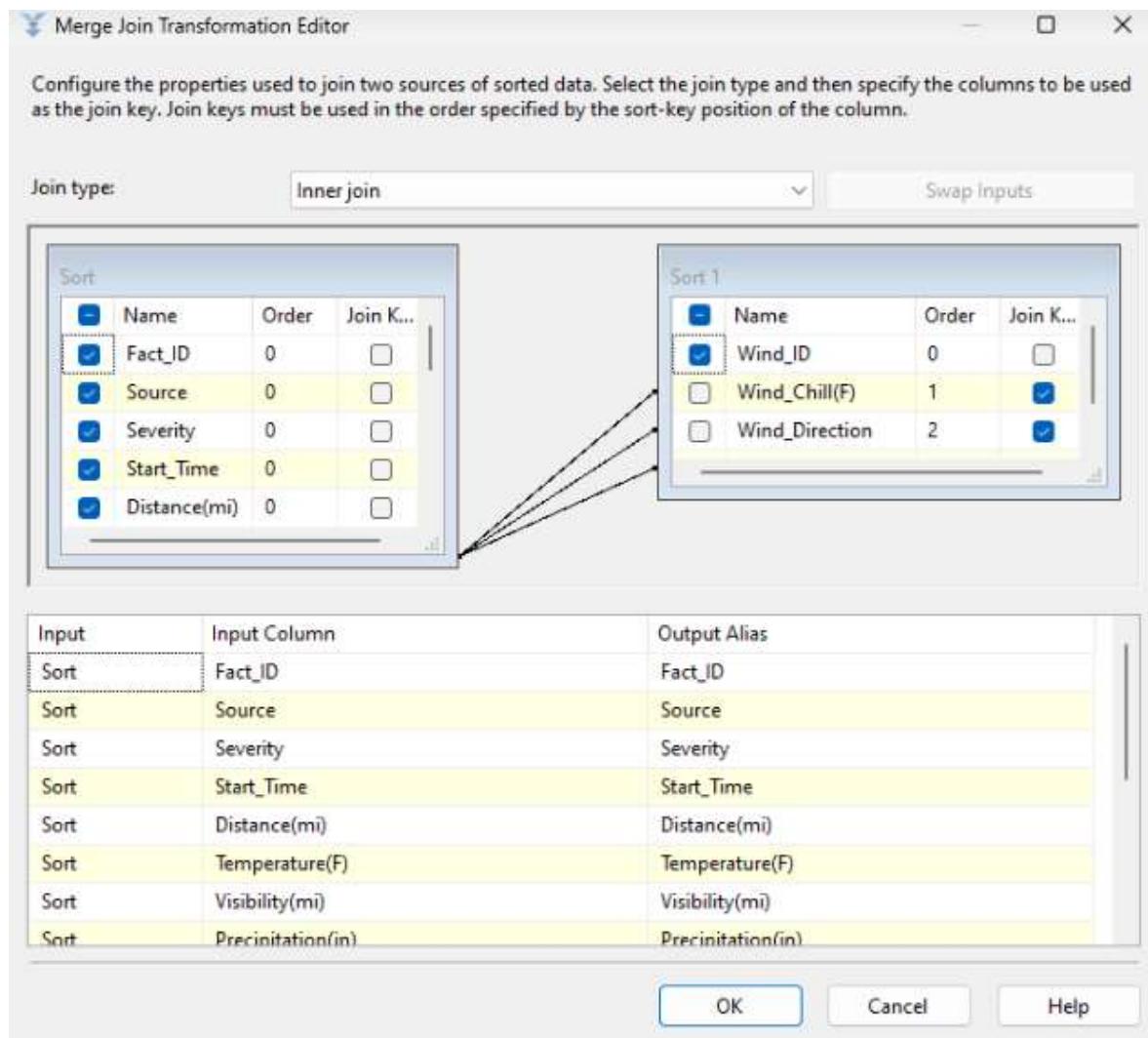
- **Bước 9.** Tương tự ta chọn cột Wind\_Direction và Wind\_Speed cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



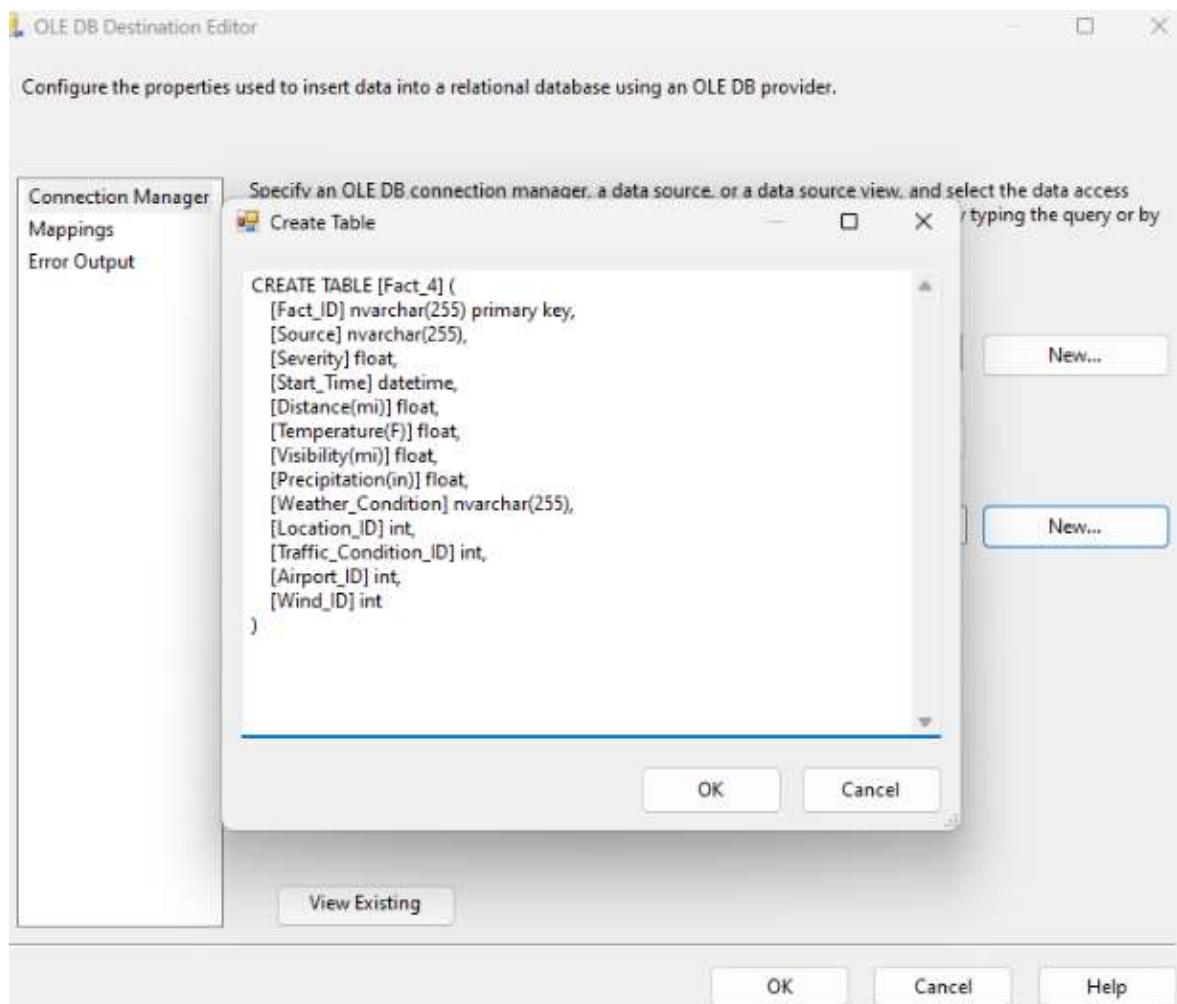
- Nối Sort1 với Merge Join
- **Bước 10.**
  - Chuột phải vào Merge Join và nhấn Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy các thuộc tính Wind\_Direction và Wind\_Speed.
    - Tiếp theo ta chọn Wind\_ID ở Sort1 để merge vào Fact3
    - Kết quả sau khi merge là bảng Fact3 không còn 2 thuộc tính Wind\_Direction, Wind\_Speed và có thêm 1 thuộc tính mới là Wind\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

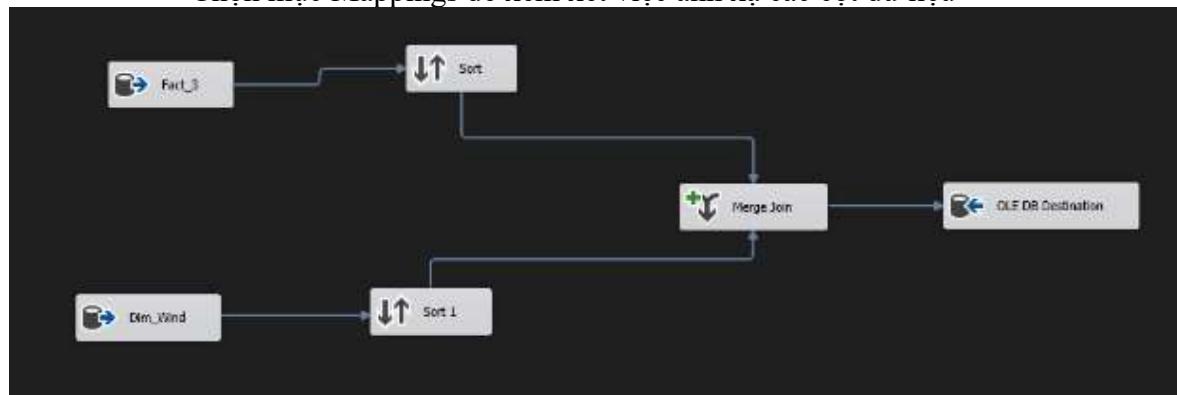


- **Bước 11.** Tạo bảng Fact4 từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

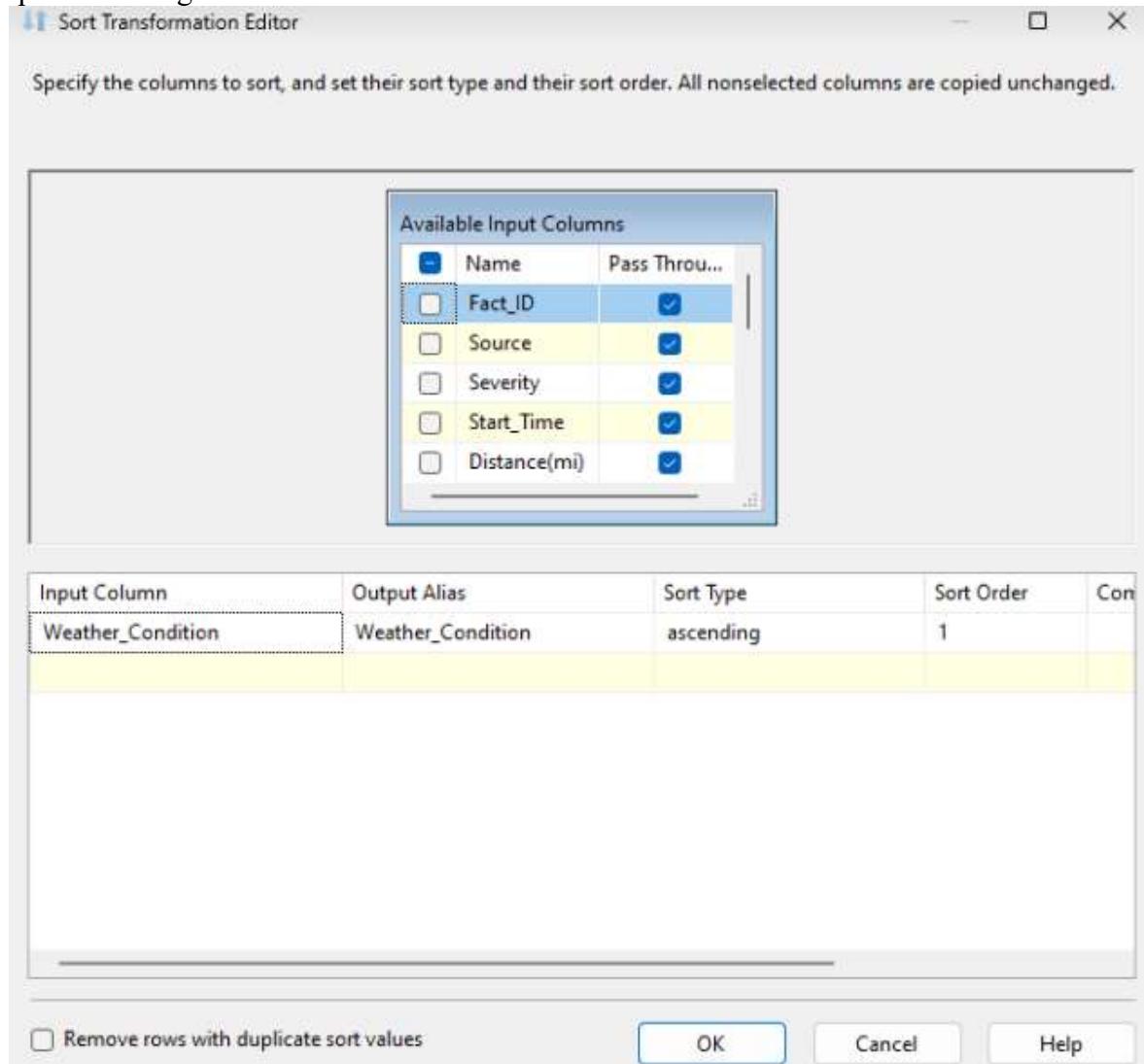


### 3.10.5. Merge Fact4 và Dim\_Weather vào Fact5

- Bước 1.** Ở tab Control Flow, tạo thêm một Data Flow Task và đổi tên Data Flow Task này là “Merge Fact4 and Dim\_Weather to Fact5”
- Bước 2.** Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact4 và Dim\_ Weather
- Bước 3.** Click chuột phải vào Fact4 chọn Edit, sau đó chọn bảng Fact3 đã được tạo khi merge Fact3 và Dim\_Wind làm data source.
- Bước 4.** Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- Bước 5.** Thực hiện chọn ánh xạ các cột cho Dim\_ Weather

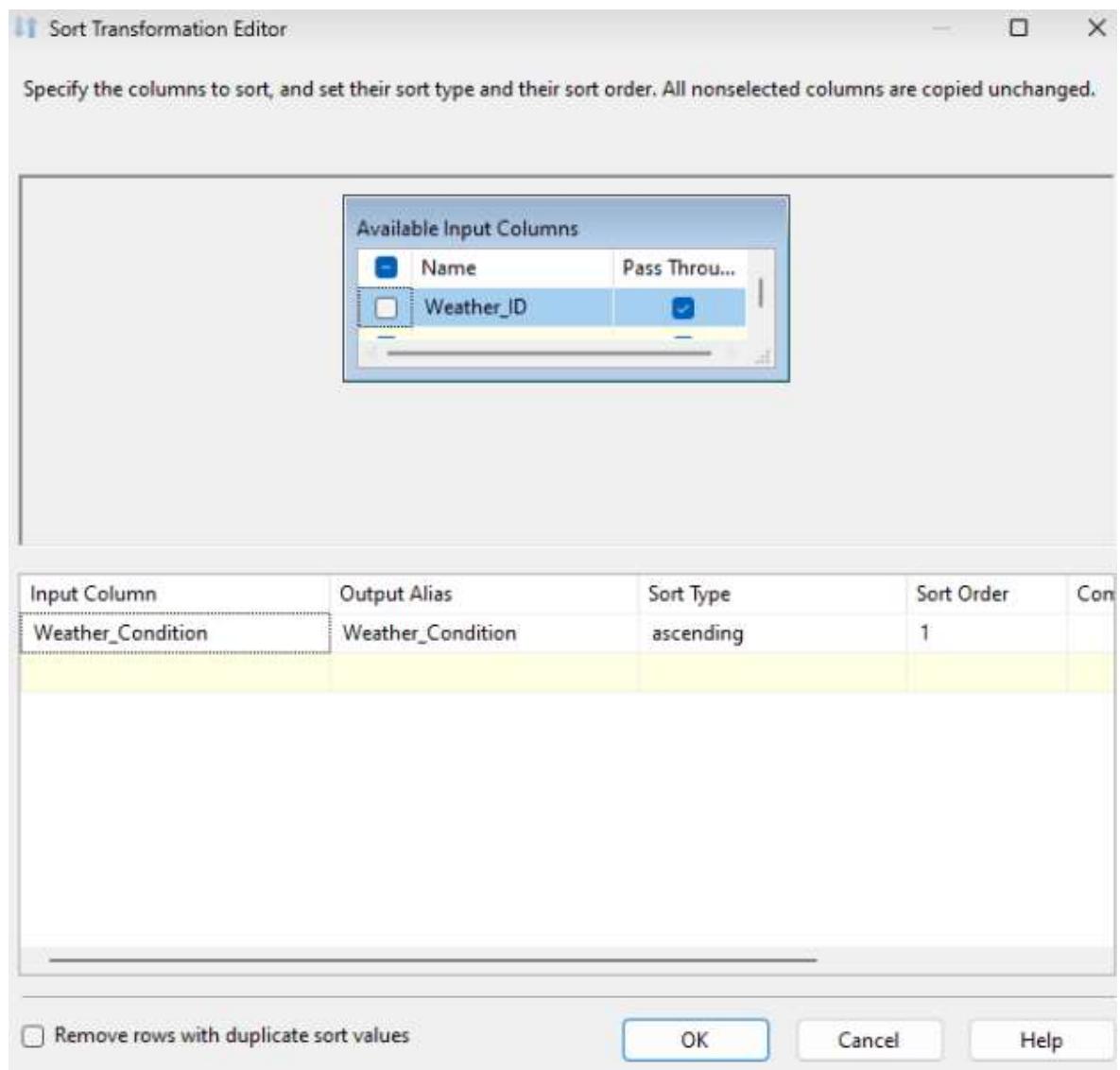
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- **Bước 6.** Tạo 2 Sort tương ứng với mỗi Source
- **Bước 7.** Tại Sort, click chuột phải chọn Edit và chọn thuộc tính Weather theo thứ tự giống với bảng Dim\_ Weather để chuẩn bị cho quá trình merge



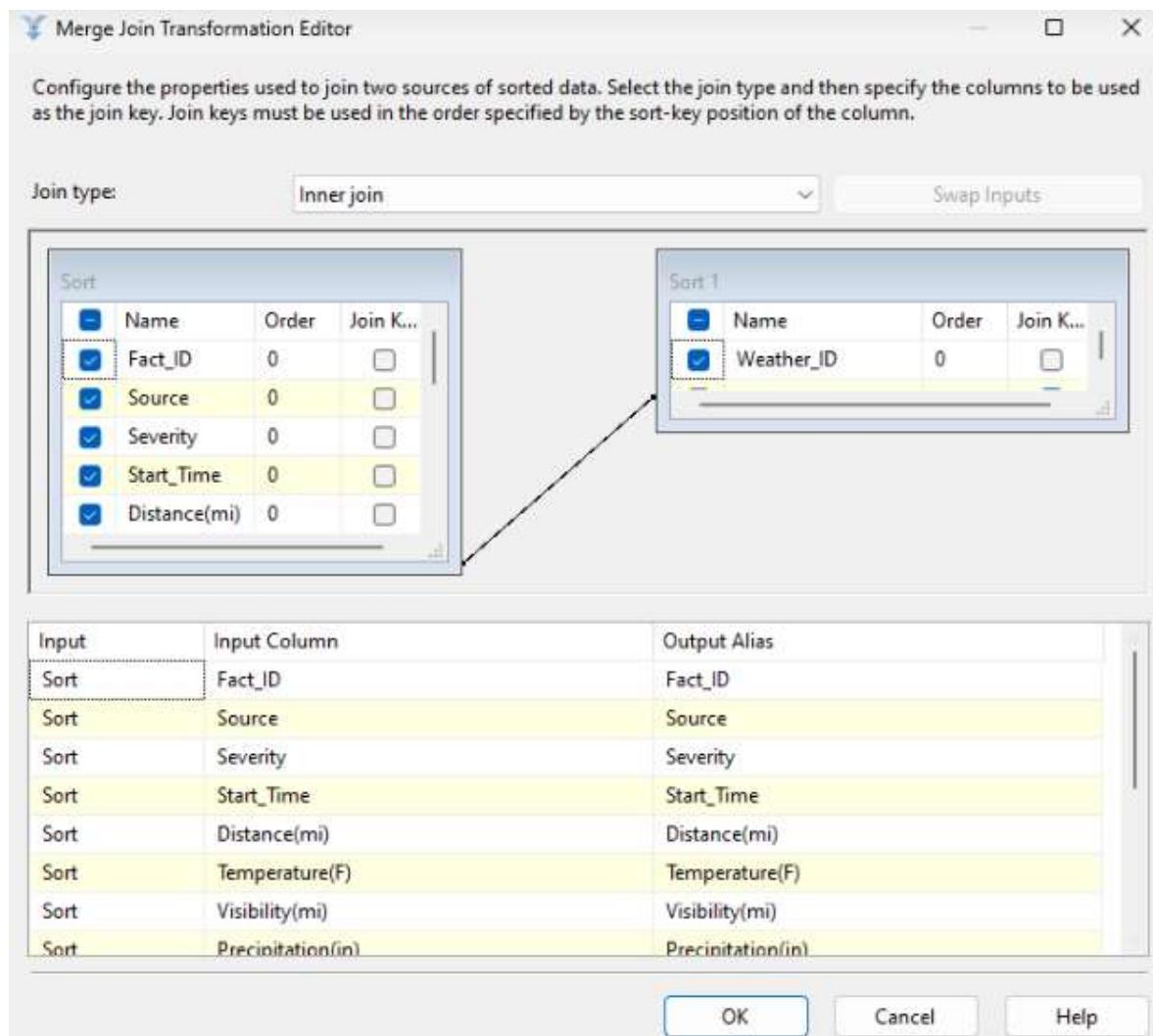
- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact4 bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_ Weather hay không.
- **Bước 9.** Tương tự ta chọn thuộc tính Weather cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



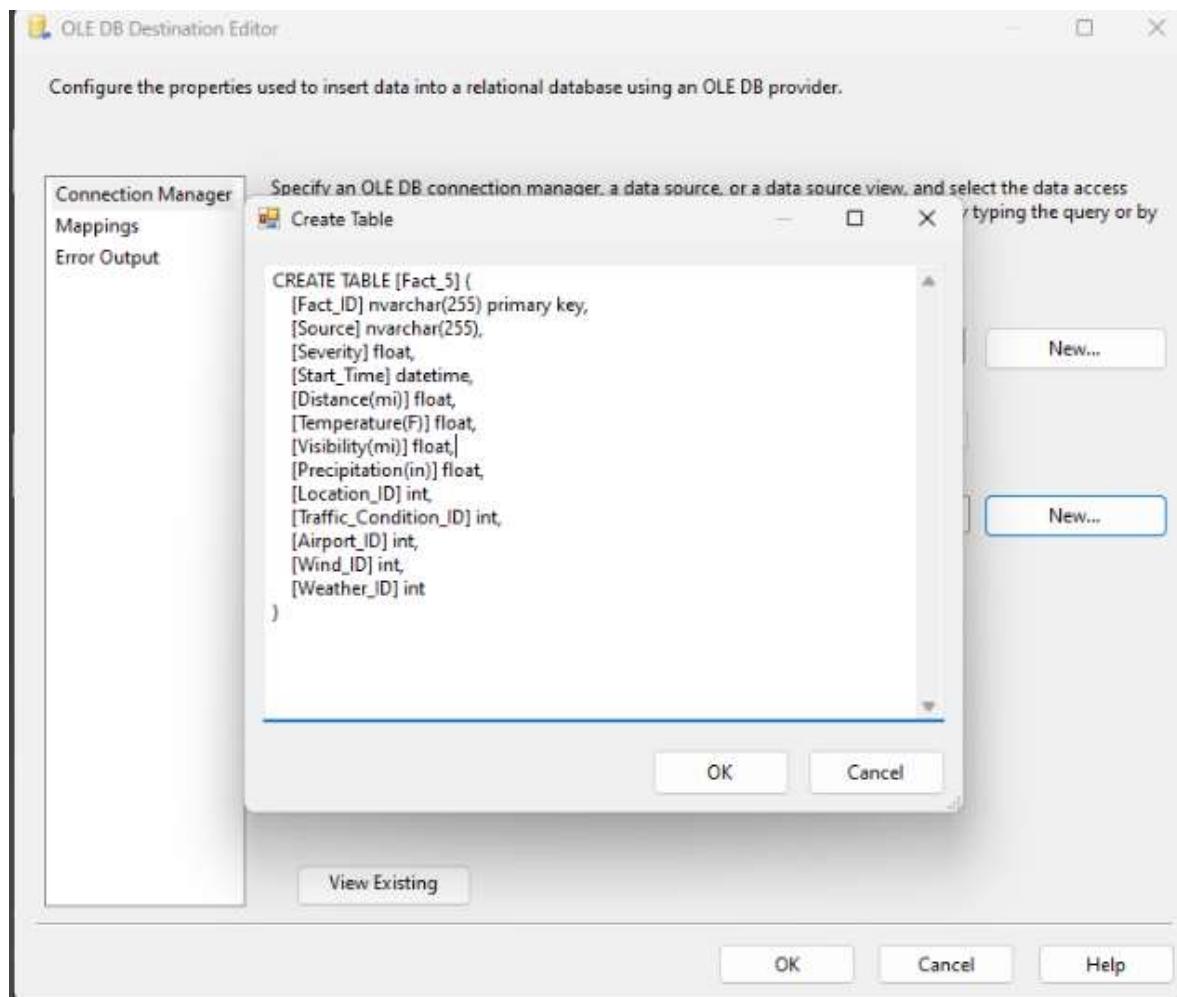
- Nối Sort1 với Merge Join
- **Bước 10.**
  - Chuột phải vào Merge Join và nhấn Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy các thuộc tính Weather.
    - Tiếp theo ta chọn Wind\_ID ở Sort1 để merge vào Fact4
    - Kết quả sau khi merge là bảng Fact4 không còn thuộc tính Weather và có thêm 1 thuộc tính mới là Weather\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

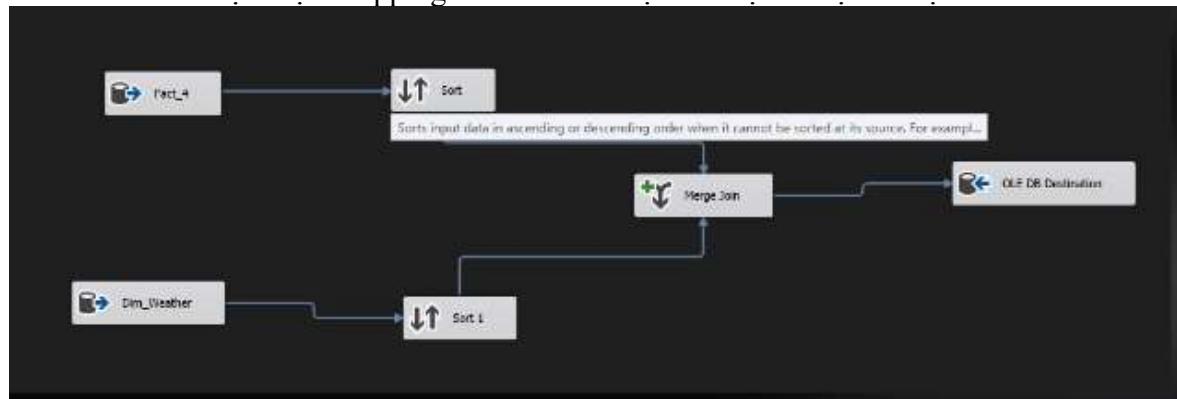


- **Bước 11.** Tạo bảng Fact5 từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

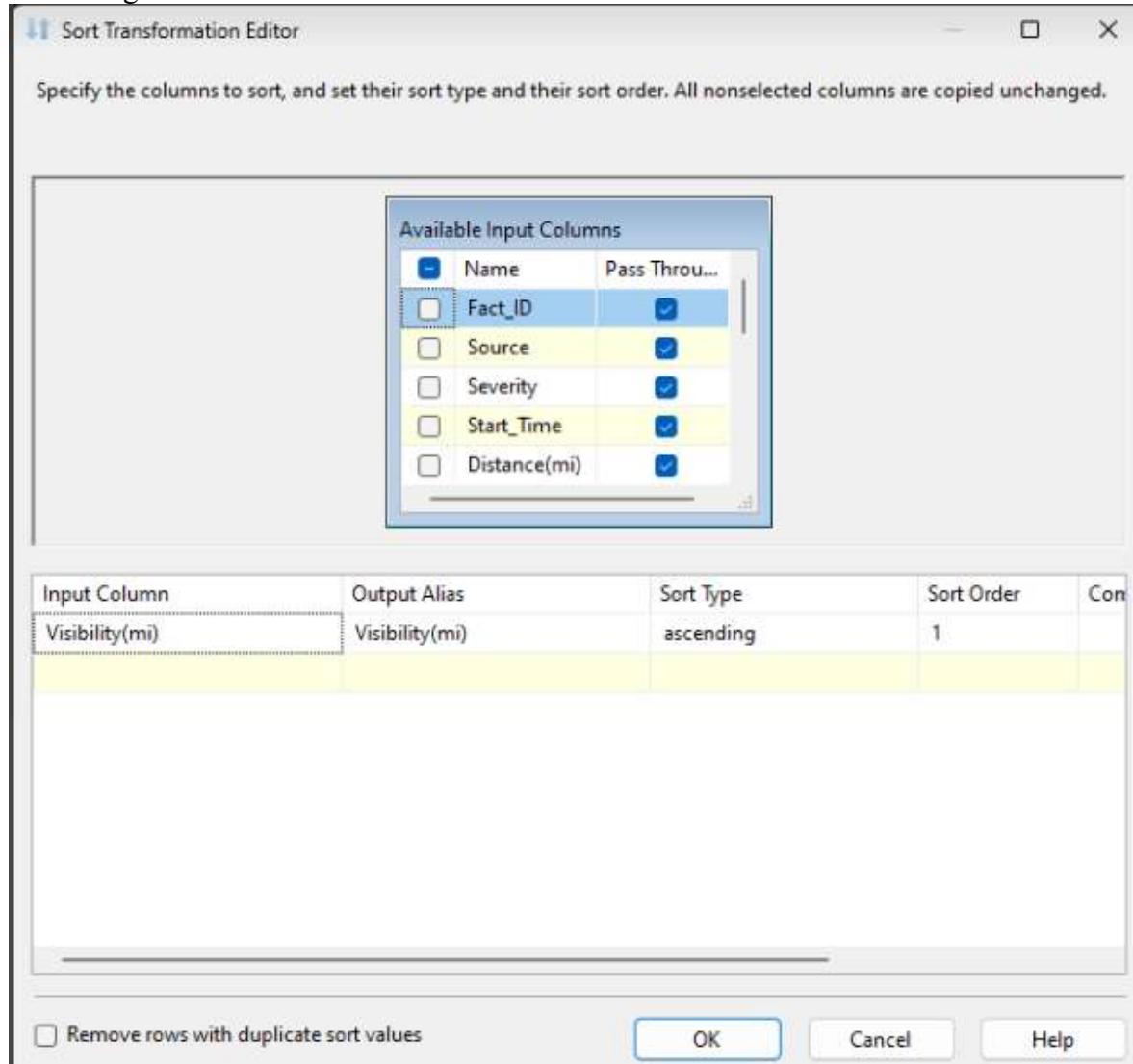


### 3.10.6. Merge Fact5 và Dim\_Visibility vào Fact6

- Bước 1.** Ở tab Control Flow, tạo thêm một Data Flow Task và đổi tên Data Flow Task này là “Merge Fact5 and Dim\_Visibility to Fact6”
- Bước 2.** Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact5 và Dim\_Visibility
- Bước 3.** Click chuột phải vào Fact5 chọn Edit, sau đó chọn bảng Fact5 đã được tạo khi merge Fact4 và Dim\_Temperature làm data source.
- Bước 4.** Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- Bước 5.** Thực hiện chọn ánh xạ các cột cho Dim\_Visibility

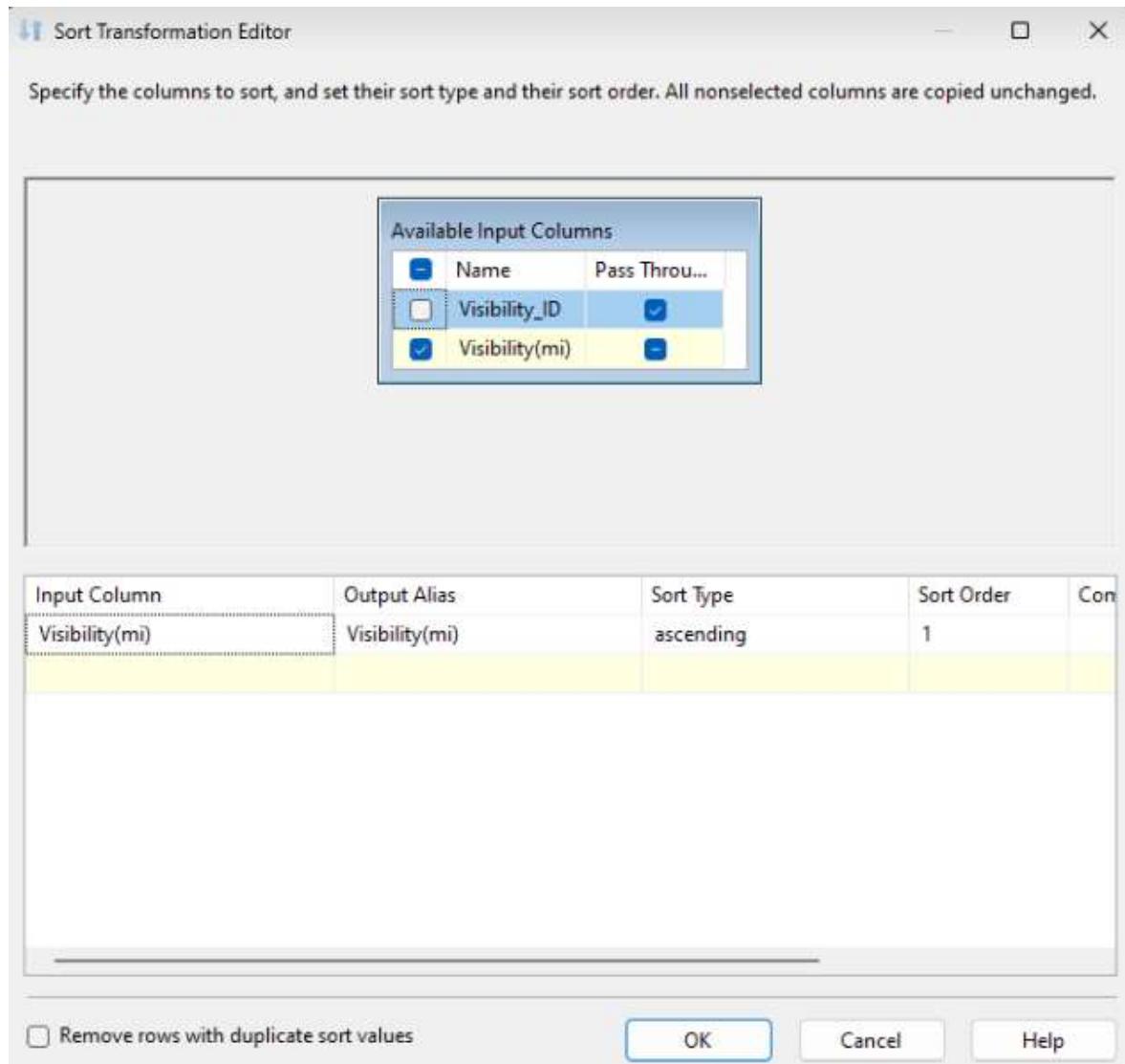
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- **Bước 6.** Tạo 2 Sort tương ứng với mỗi Source
- **Bước 7.** Tại Sort, click chuột phải chọn Edit và chọn thuộc tính Visibility(mi) theo thứ tự giống với bảng Dim\_Visibility để chuẩn bị cho quá trình merge



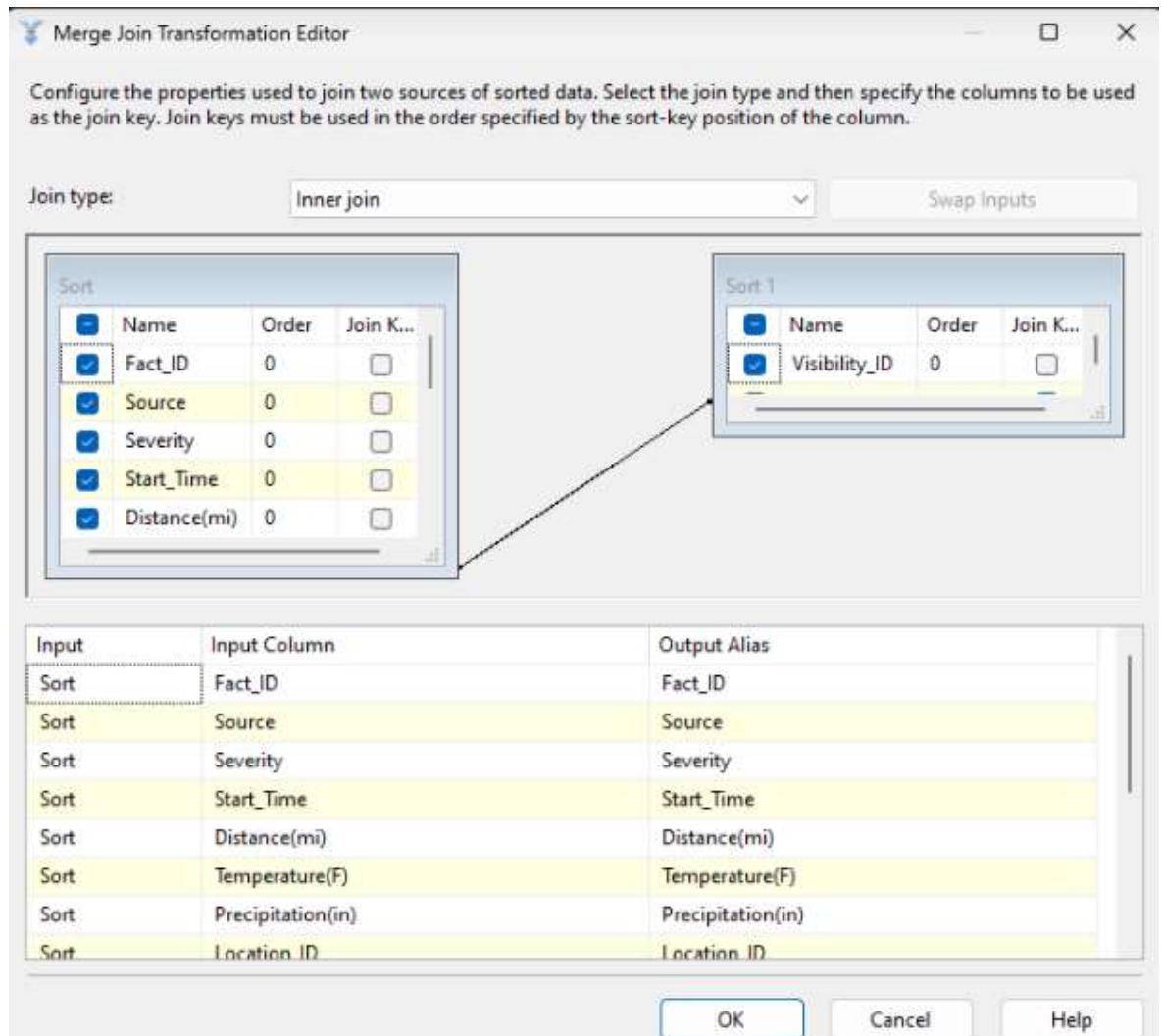
- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact5 bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_Visibility hay không.
- **Bước 9.** Tương tự ta chọn thuộc tính Visibility(mi) cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



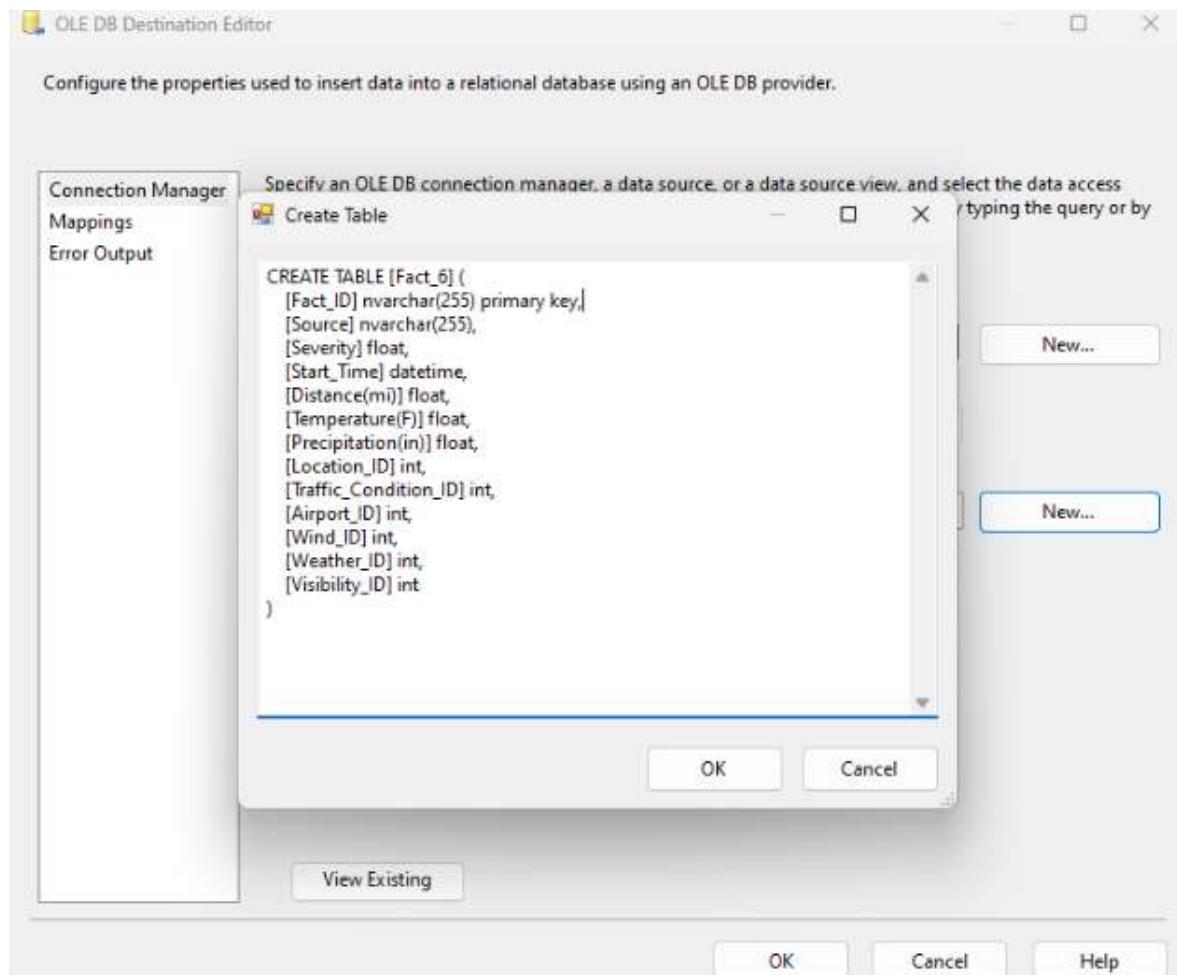
- Nối Sort1 với Merge Join
- **Bước 10.**
  - Chuột phải vào Merge Join và nhấn Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy các thuộc tính Visibility(mi).
    - Tiếp theo ta chọn Wind\_ID ở Sort1 để merge vào Fact5
    - Kết quả sau khi merge là bảng Fact5 không còn thuộc tính Visibility(mi) và có thêm 1 thuộc tính mới là Visibility\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

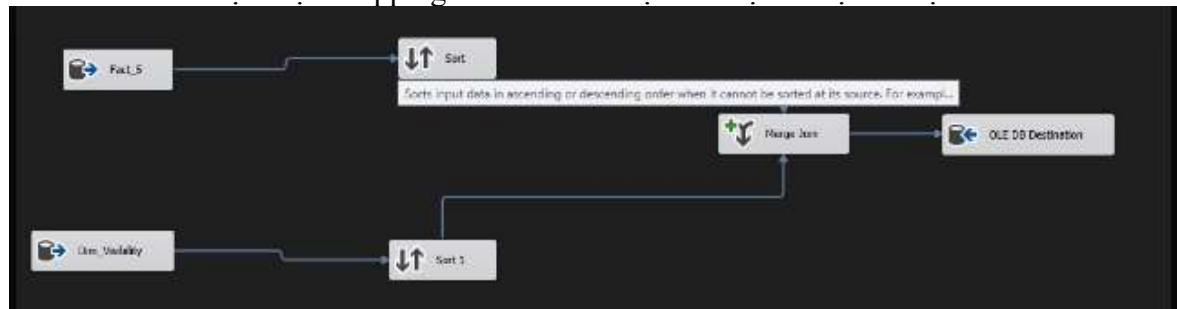


- **Bước 11.** Tạo bảng Fact6 từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

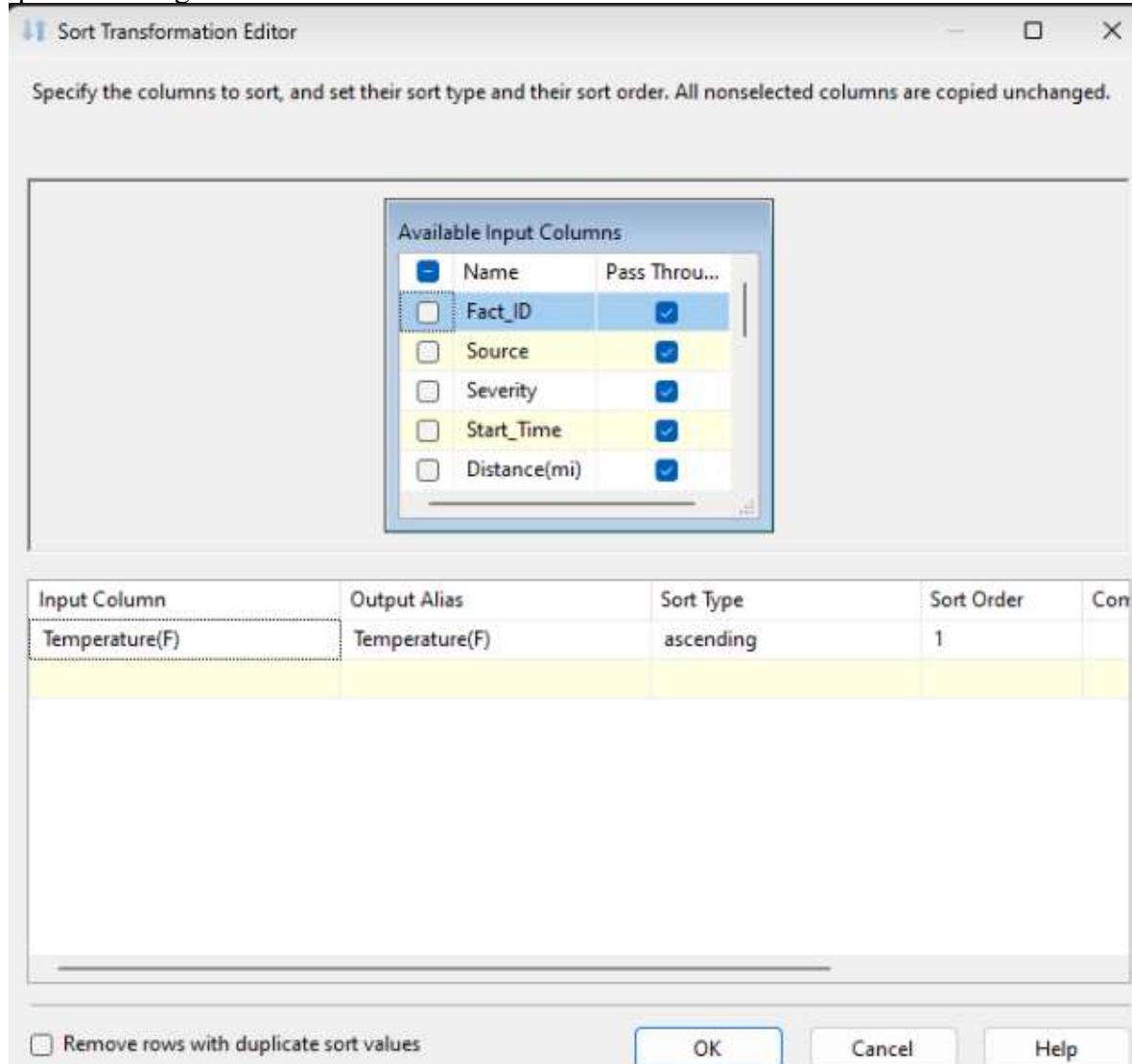


### 3.10.7. Merge Fact6 và Dim\_Temperature vào Fact7

- Bước 1. Ở tab Control Flow, tạo thêm một Data Flow Task và đổi tên Data Flow Task này là “Merge Fact6 and Dim\_Temperature to Fact7”
- Bước 2. Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact6 và Dim\_Temperature
- Bước 3. Click chuột phải vào Fact6 chọn Edit, sau đó chọn bảng Fact6 đã được tạo khi merge Fact5 và Dim\_Visibility làm data source.
- Bước 4. Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- Bước 5. Thực hiện chọn ánh xạ các cột cho Dim\_Temperature
  - Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- Bước 6. Tạo 2 Sort tương ứng với mỗi Source
- Bước 7. Tại Sort, click chuột phải chọn Edit và chọn thuộc tính

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

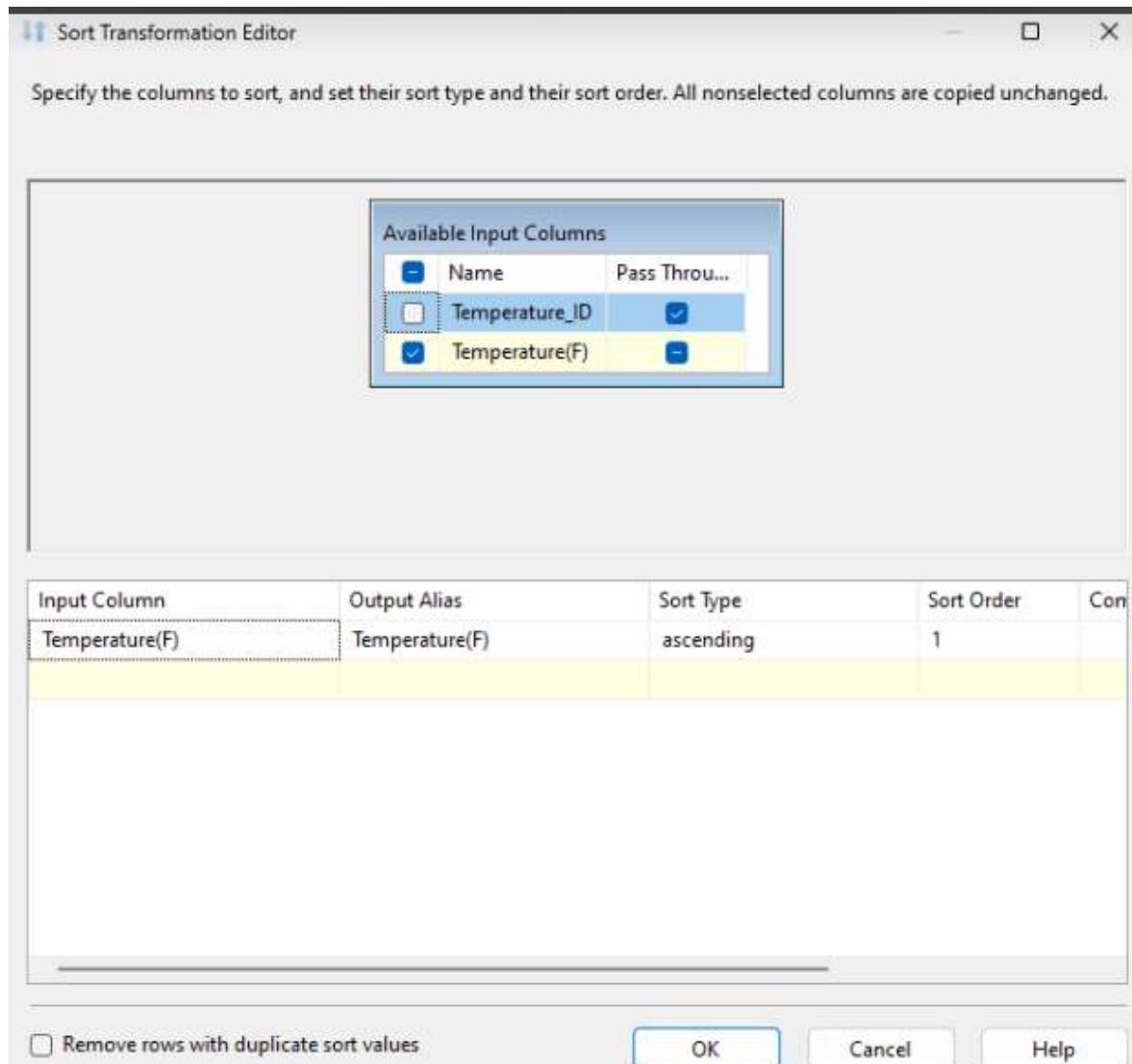
Temperature theo thứ tự giống với bảng Dim\_Temperature để chuẩn bị cho quá trình merge



- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact6 bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_Temperature hay không.

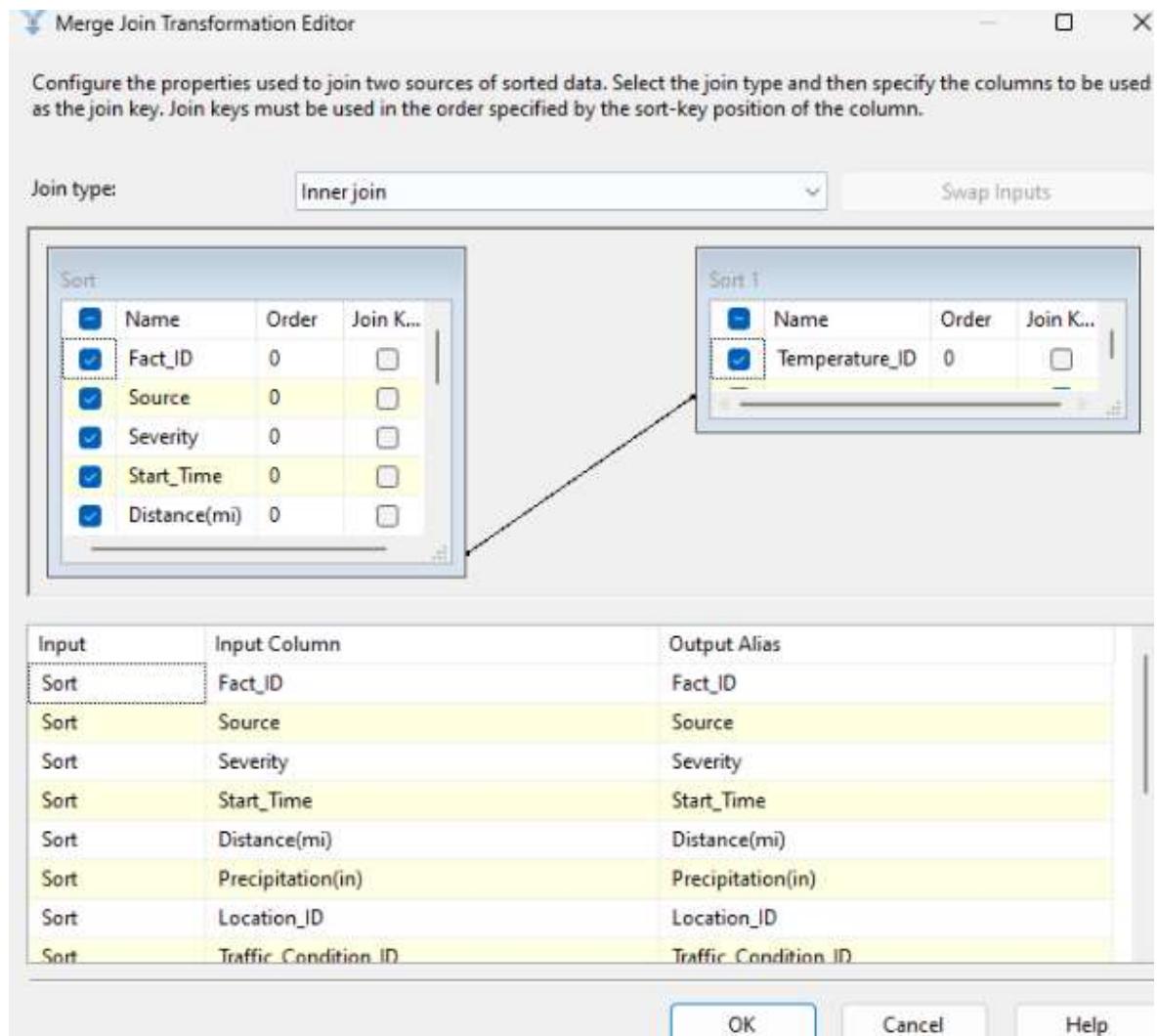
- **Bước 9.** Tương tự ta chọn thuộc tính Temperature cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



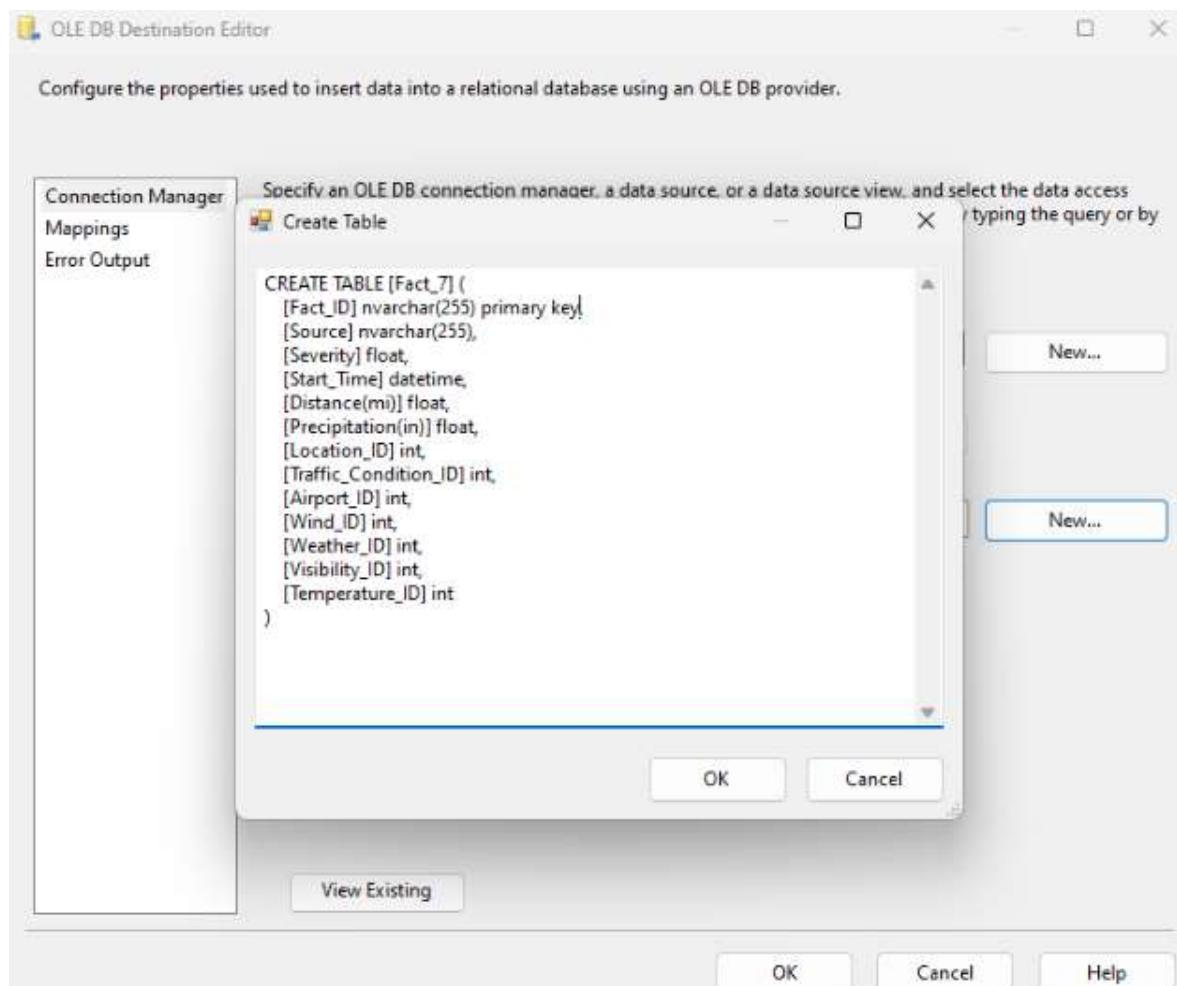
- Nối Sort1 với Merge Join
- **Bước 10.**
  - Chuột phải vào Merge Join và nhấp Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy các thuộc tính Temperature.
    - Tiếp theo ta chọn Temperature\_ID ở Sort1 để merge vào Fact6
    - Kết quả sau khi merge là bảng Fact6 không còn thuộc tính Temperature và có thêm 1 thuộc tính mới là Temperature\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

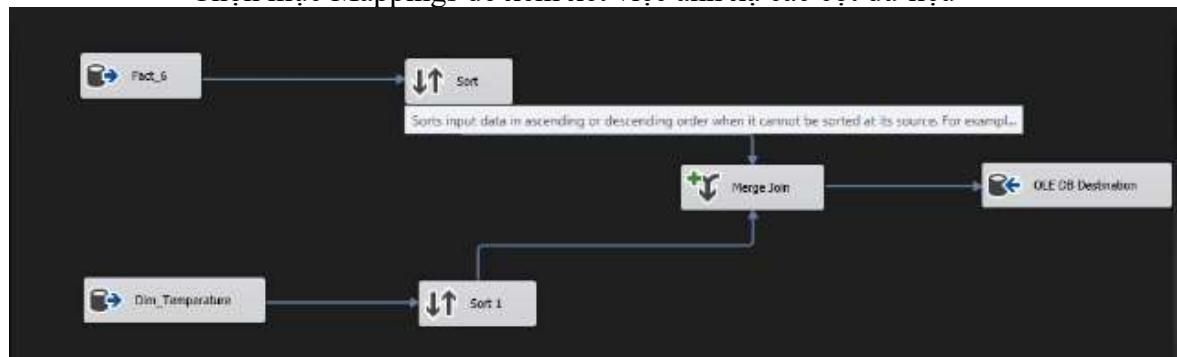


- **Bước 11.** Tạo bảng Fact7 từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu

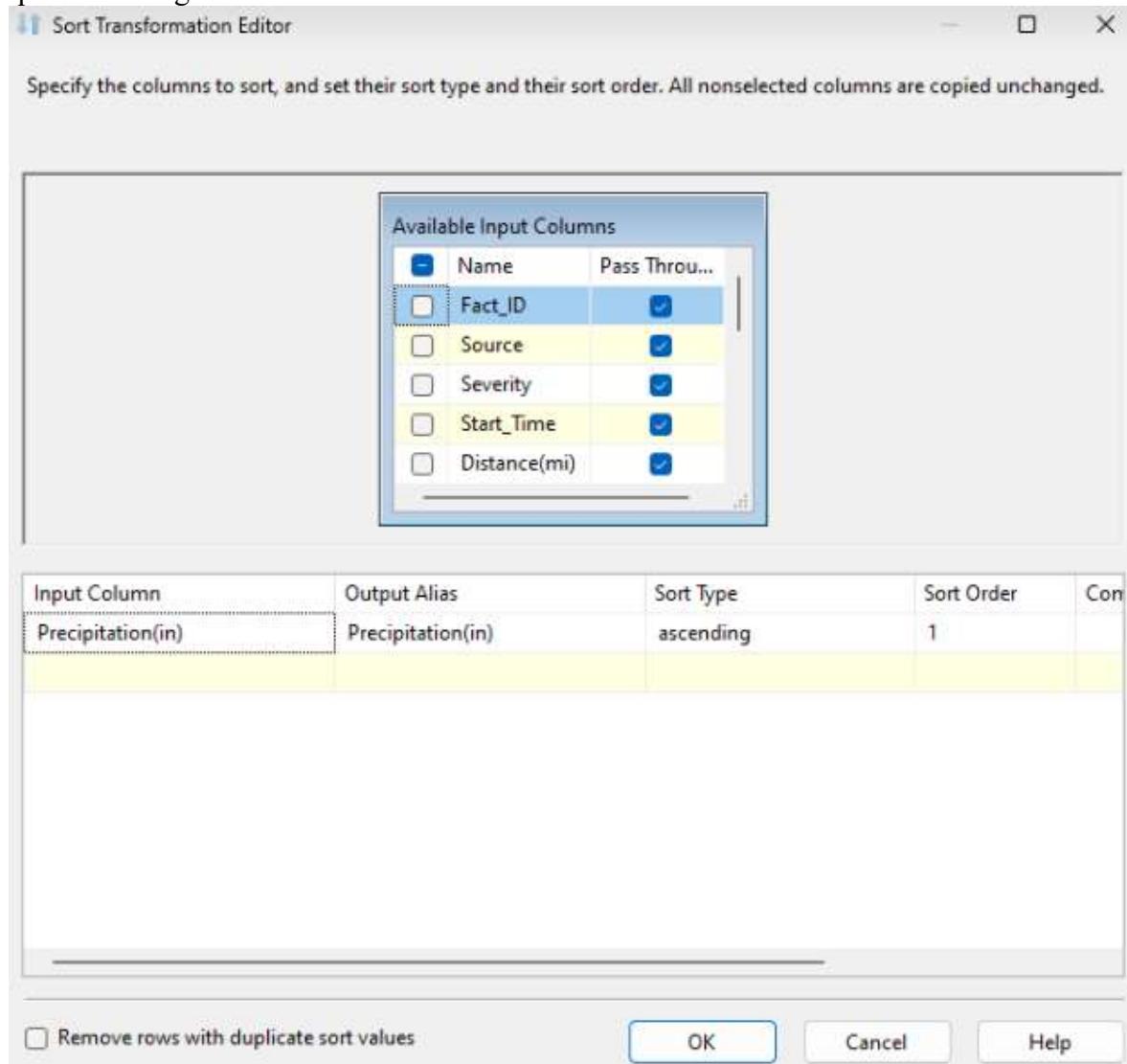


### 3.10.8.Merge Fact7 và Dim\_Precipitation vào Fact

- Bước 1.** Ở tab Control Flow, tạo thêm một Data Flow Task và đổi tên Data Flow Task này là “Merge Fact7 and Dim\_Precipitation to Fact”
- Bước 2.** Click chuột phải vào Data Flow Task nói trên và chọn Edit, trong tab Data Flow ta tạo 2 OLE DB Source và đổi tên Fact7 và Dim\_Precipitation
- Bước 3.** Click chuột phải vào Fact7 chọn Edit, sau đó chọn bảng Fact7 đã được tạo khi merge Fact6 và Dim\_Temperature làm data source.
- Bước 4.** Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- Bước 5.** Thực hiện chọn ánh xạ các cột cho Dim\_Precipitation

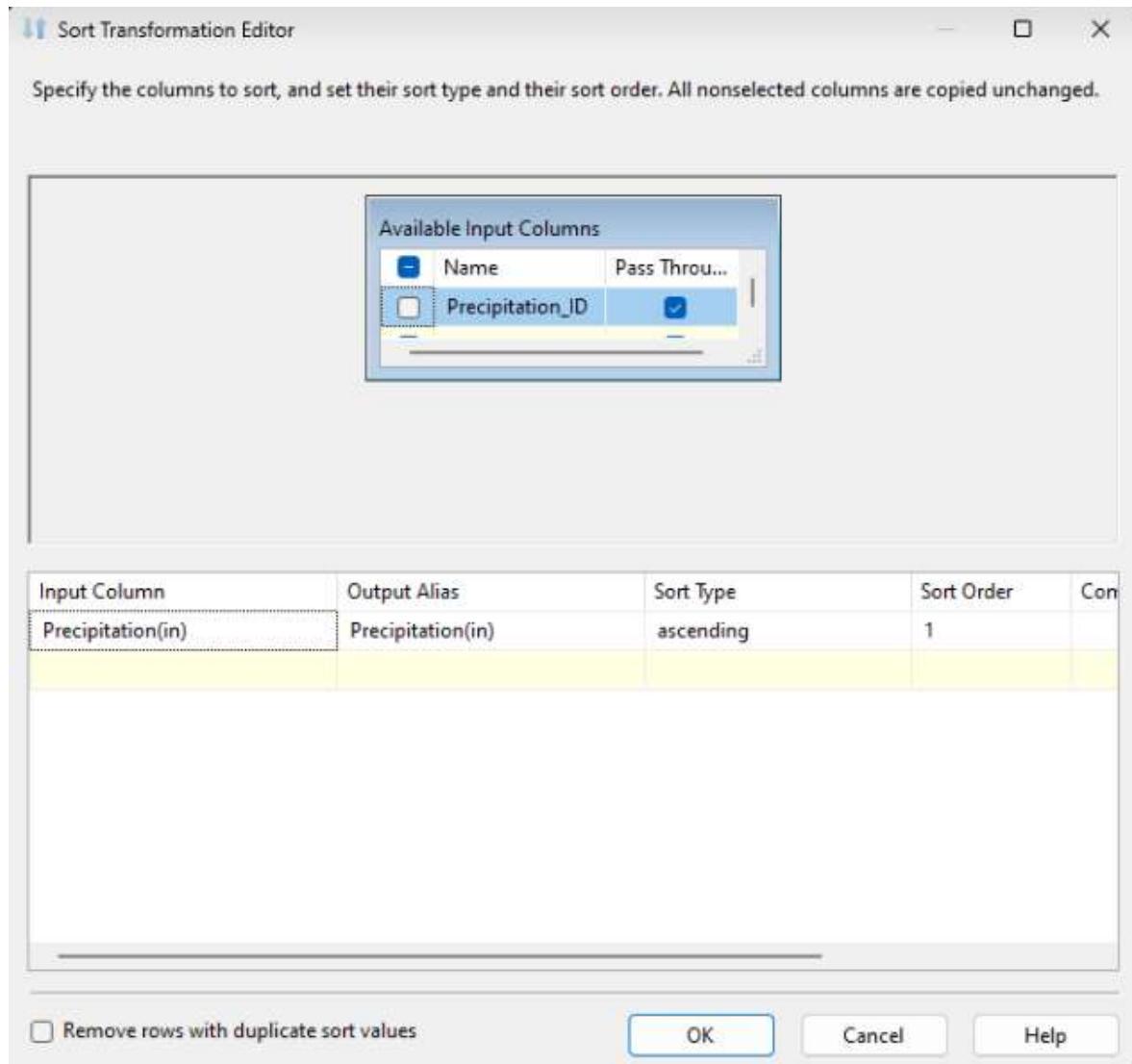
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- Chọn mục Columns để xem xét các cột được ánh xạ. Nhấn OK.
- **Bước 6.** Tạo 2 Sort tương ứng với mỗi Source
- **Bước 7.** Tại Sort, click chuột phải chọn Edit và chọn thuộc tính Weather\_Condition theo thứ tự giống với bảng Dim\_Precipitation để chuẩn bị cho quá trình merge



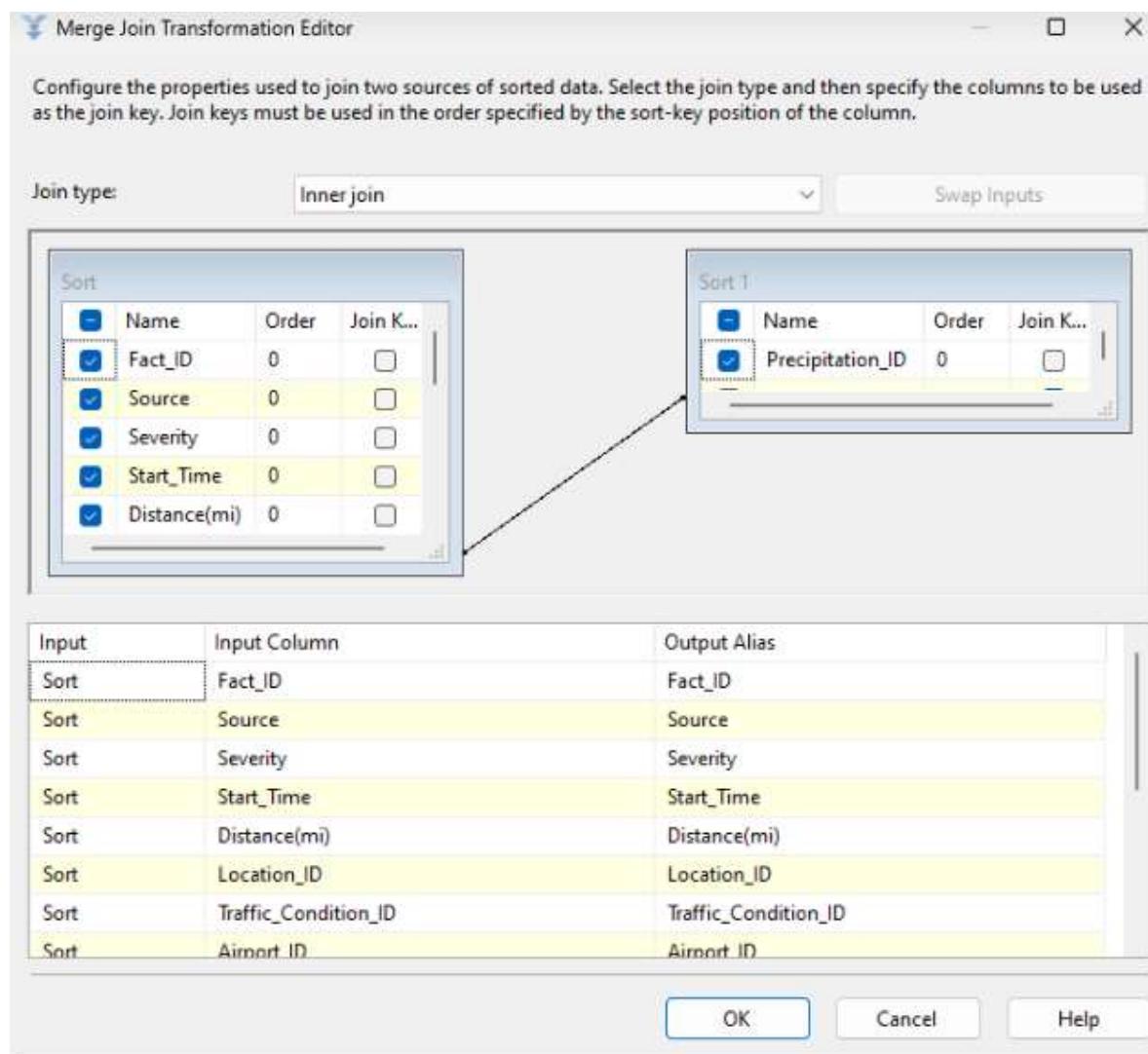
- **Bước 8.** Tạo một Merge Join và nối với Sort, tiếp theo chọn Merge Join Left Input để giữ lại toàn bộ các dòng trong bảng Fact7 bất kể có kết quả khi thực hiện phép kết trái với cột ID của bảng Dim\_Precipitation hay không.
- **Bước 9.** Tương tự ta chọn thuộc tính Precipitation cho Sort1

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



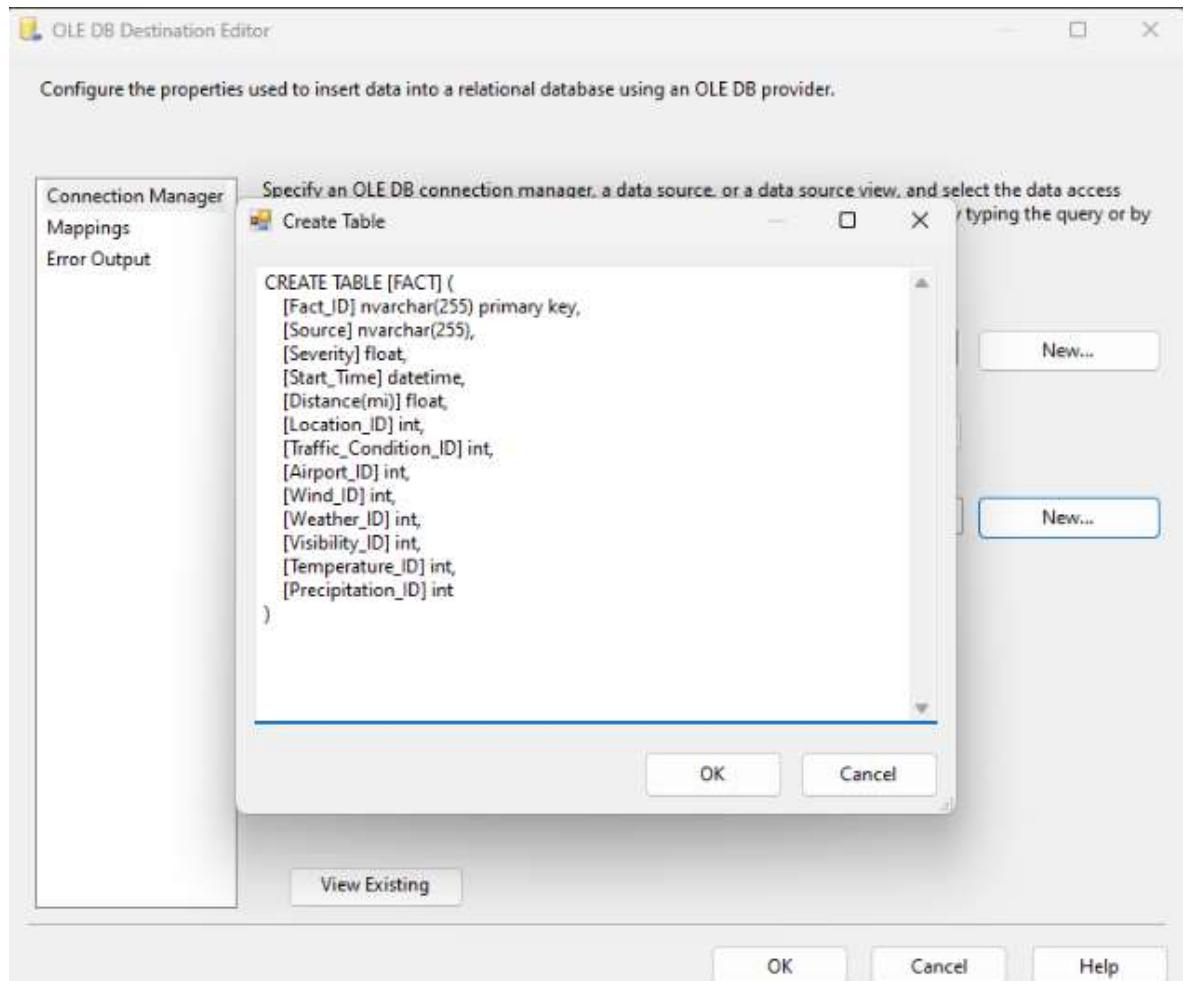
- Nối Sort1 với Merge Join
- **Bước 10.**
  - Chuột phải vào Merge Join và nhấp Edit, một hộp thoại merge editor xuất hiện: ở đây ta tick chọn tất cả các cột của Sort nhưng không lấy thuộc tính Precipitation.
    - Tiếp theo ta chọn Precipitation\_ID ở Sort1 để merge vào Fact7
    - Kết quả sau khi merge là bảng Fact7 không còn thuộc tính Precipitation và có thêm 1 thuộc tính mới là Precipitation\_ID

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 11.** Tạo bảng Fact từ một OLE DB Destination để chứa tất cả những gì đã merge

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

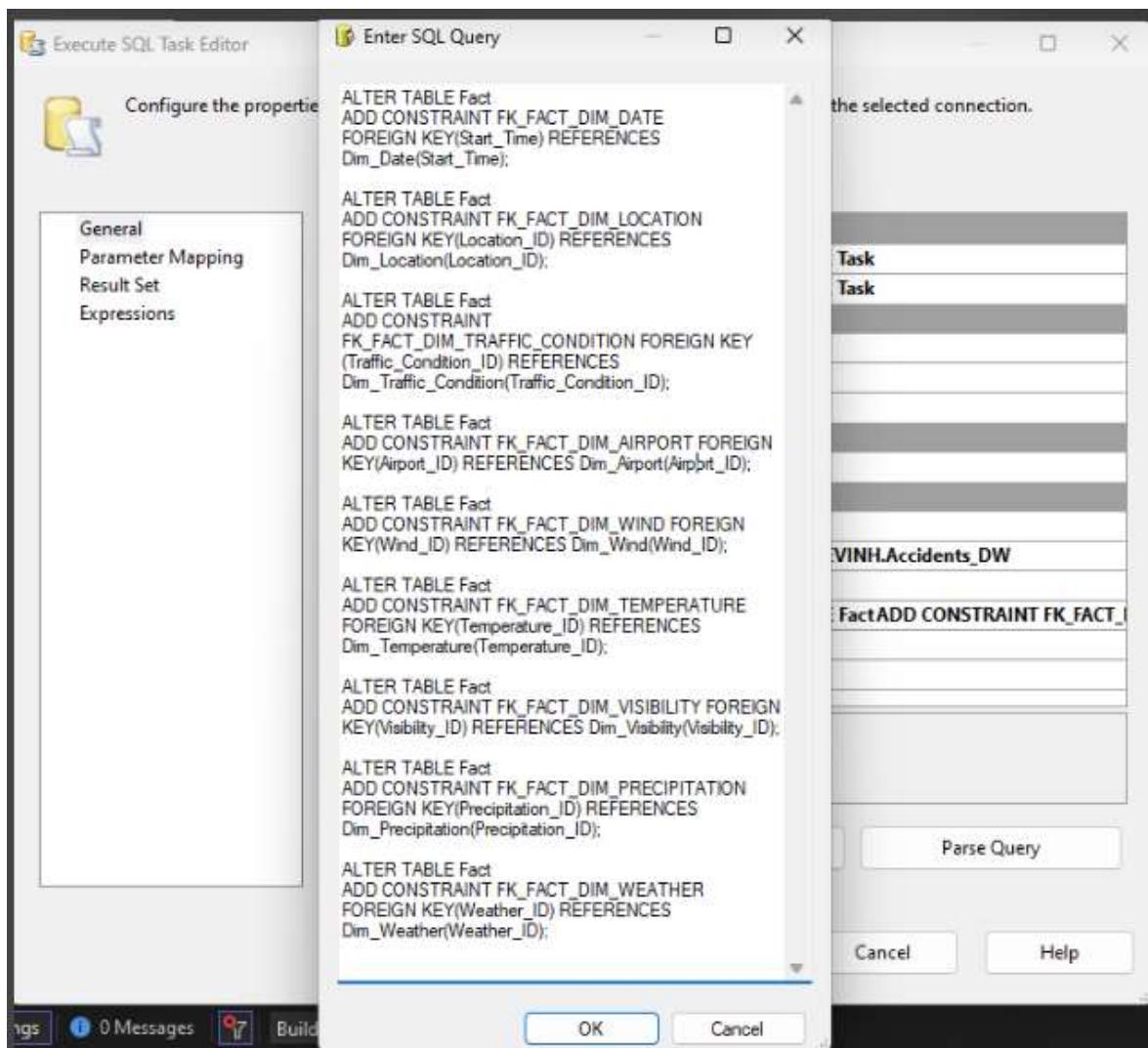


- Chọn mục Mappings để xem xét việc ánh xạ các cột dữ liệu



## Đồ án xây dựng kho dữ liệu US ACCIDENTS

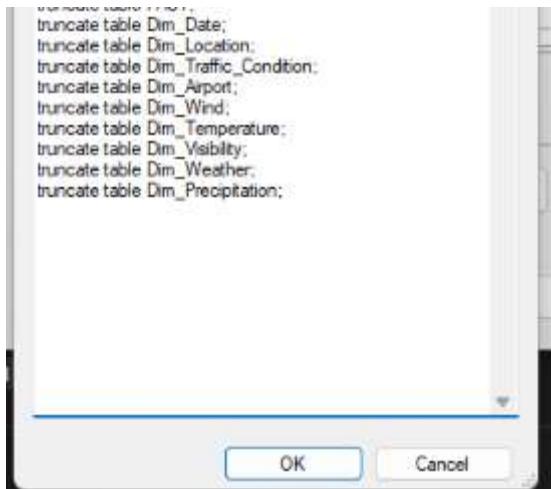
### 3.10.9. Tạo khóa ngoại từ bảng Fact đến các Dimension



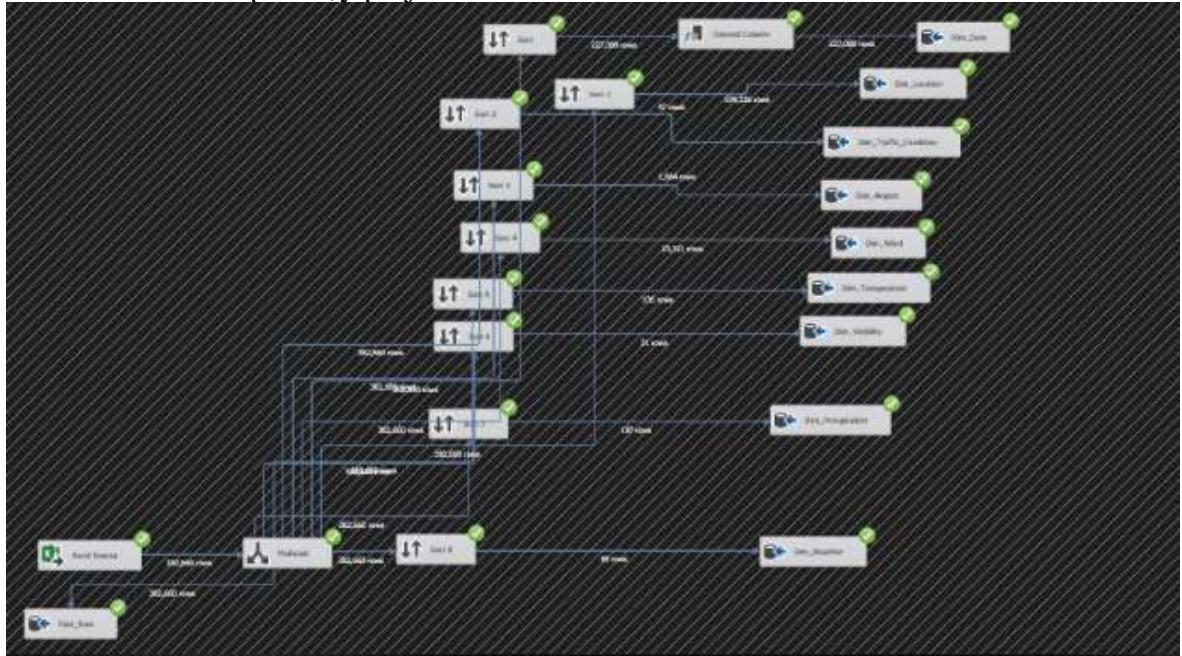
## 4. CHẠY SSIS

- **Bước 1.** Thêm vào một Execute SQL Task nhằm thực hiện nhiệm vụ đảm bảo đỗ dữ liệu mới hoàn toàn (không bị chồng chéo dữ liệu cũ) mỗi khi chạy project, trước quá trình chia bảng Fact và các Dimension
  - **Bước 2.** Nhấn chuột phải vào Execute SQL Task này và chọn Edit. Ở ô Connection, chọn connection đã thiết lập đến data warehouse trong SQL Server
  - **Bước 3.** Ở ô SQLStatement, thêm các câu truy vấn SQL thực hiện xóa dữ liệu cũ trong các bảng mỗi khi chạy project.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



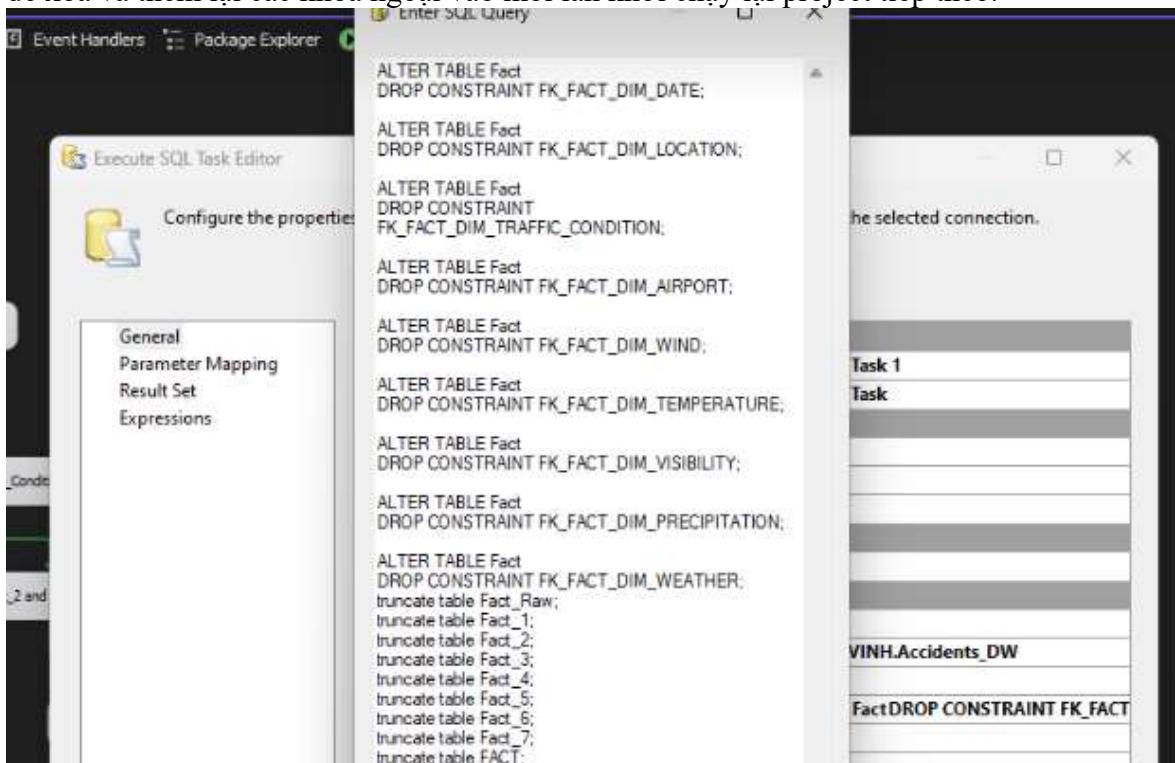
- Nhấn OK để hoàn tất quá trình.
- Bước 4. Nhấn nút Start trên thanh menu để tiến hành chạy project
  - Kết quả chạy project:



## Đồ án xây dựng kho dữ liệu US ACCIDENTS

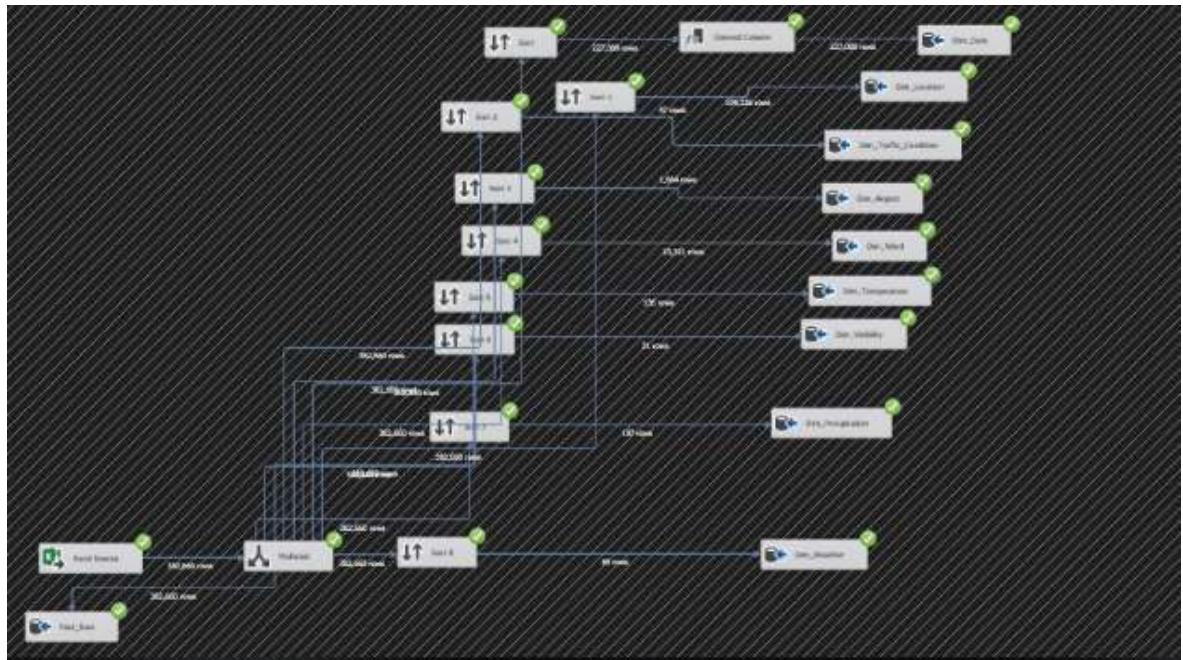


- **Bước 5:** Sau khi đã chạy project lần đầu thành công, khóa ngoại của bảng Fact tham chiếu đến các Dimension đã được tạo. Ta tiến hành thêm các lệnh SQL để xóa và thêm lại các khóa ngoại vào mỗi lần khởi chạy lại project tiếp theo.



- **Bước 6.** Tiến hành chạy lại project

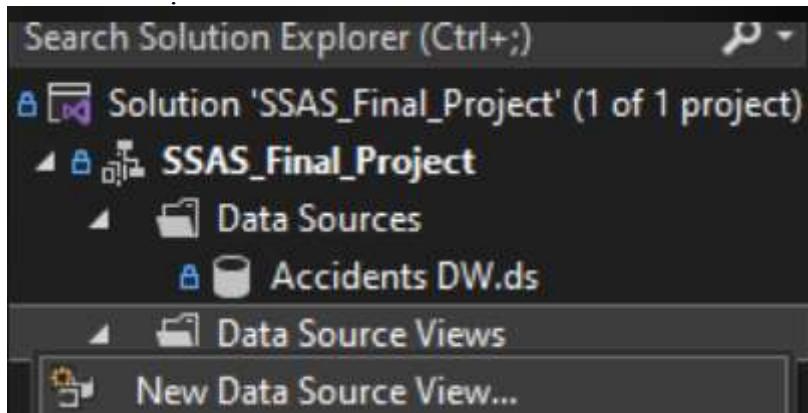
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



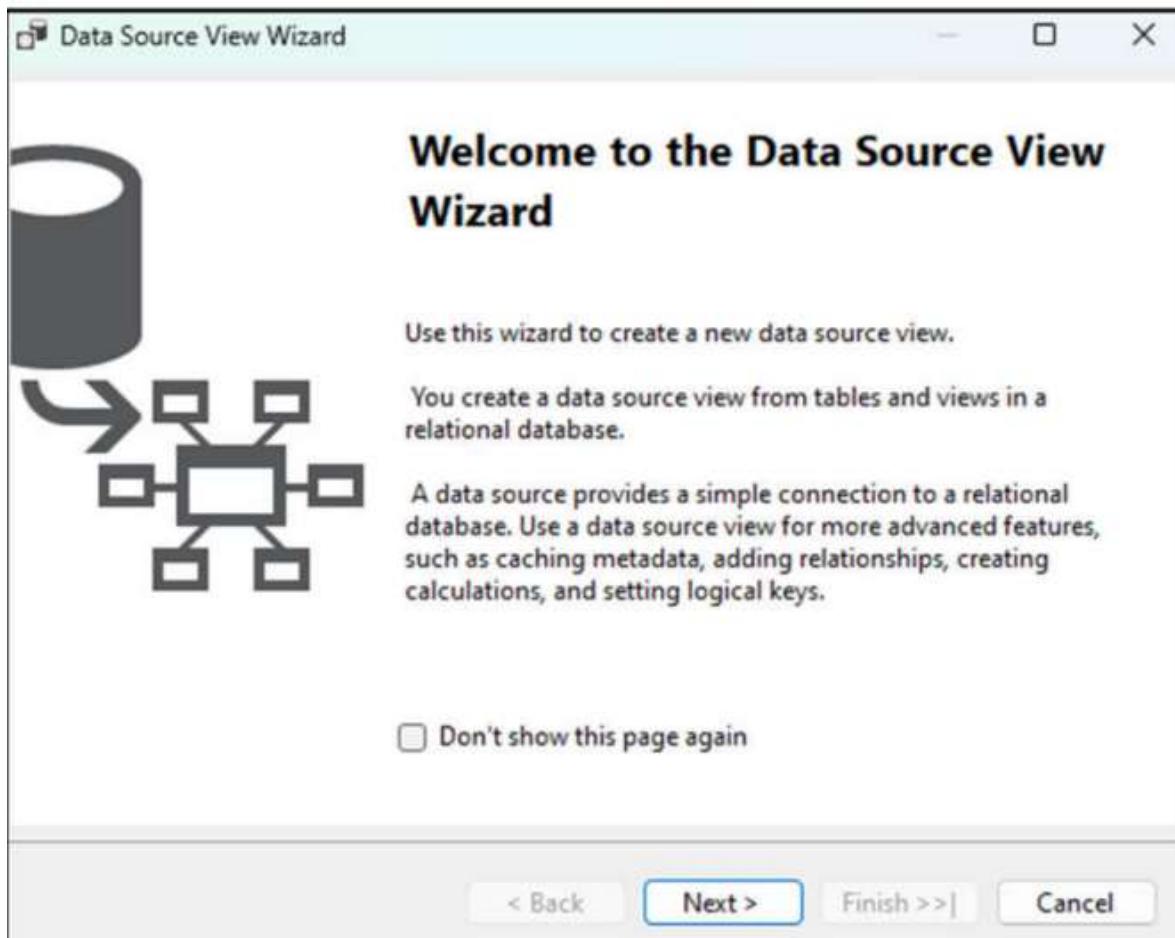
## CHƯƠNG 3. PHÂN TÍCH DỮ LIỆU TRONG KHO (SSAS)

### 1. XÁC ĐỊNH KHUNG NHÌN DỮ LIỆU NGUỒN (DEFINE DATE SOURCE VIEW)

- **Bước 1:** Tại Solution Explorer, ta click chuột phải vào thư mục Data Source Views và chọn New Data Source View.

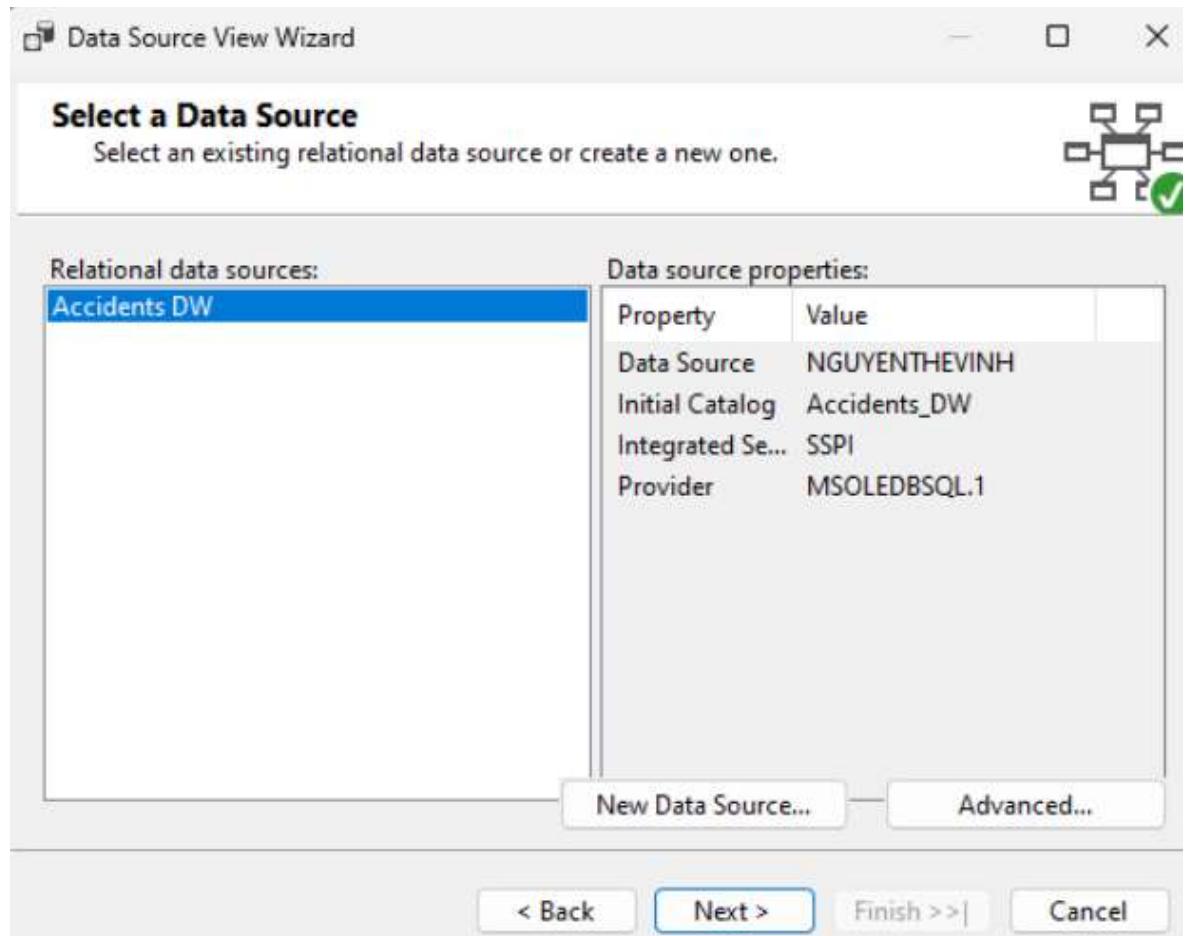


- **Bước 2:** Hộp thoại Data Source View Wizard xuất hiện, chọn Next để tiếp tục.



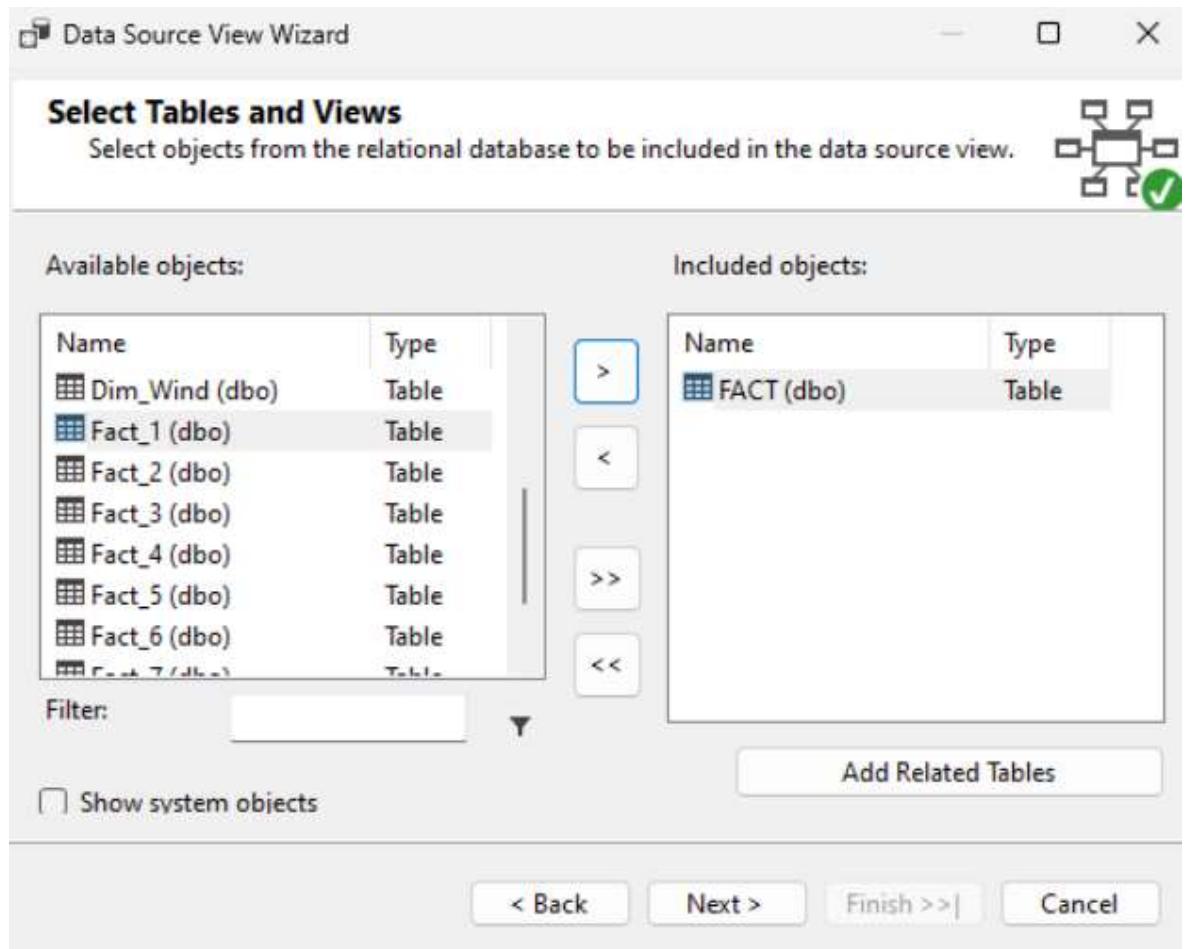
- **Bước 3:** Chọn data source vừa tạo, sau đó chọn Next để tiếp tục.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



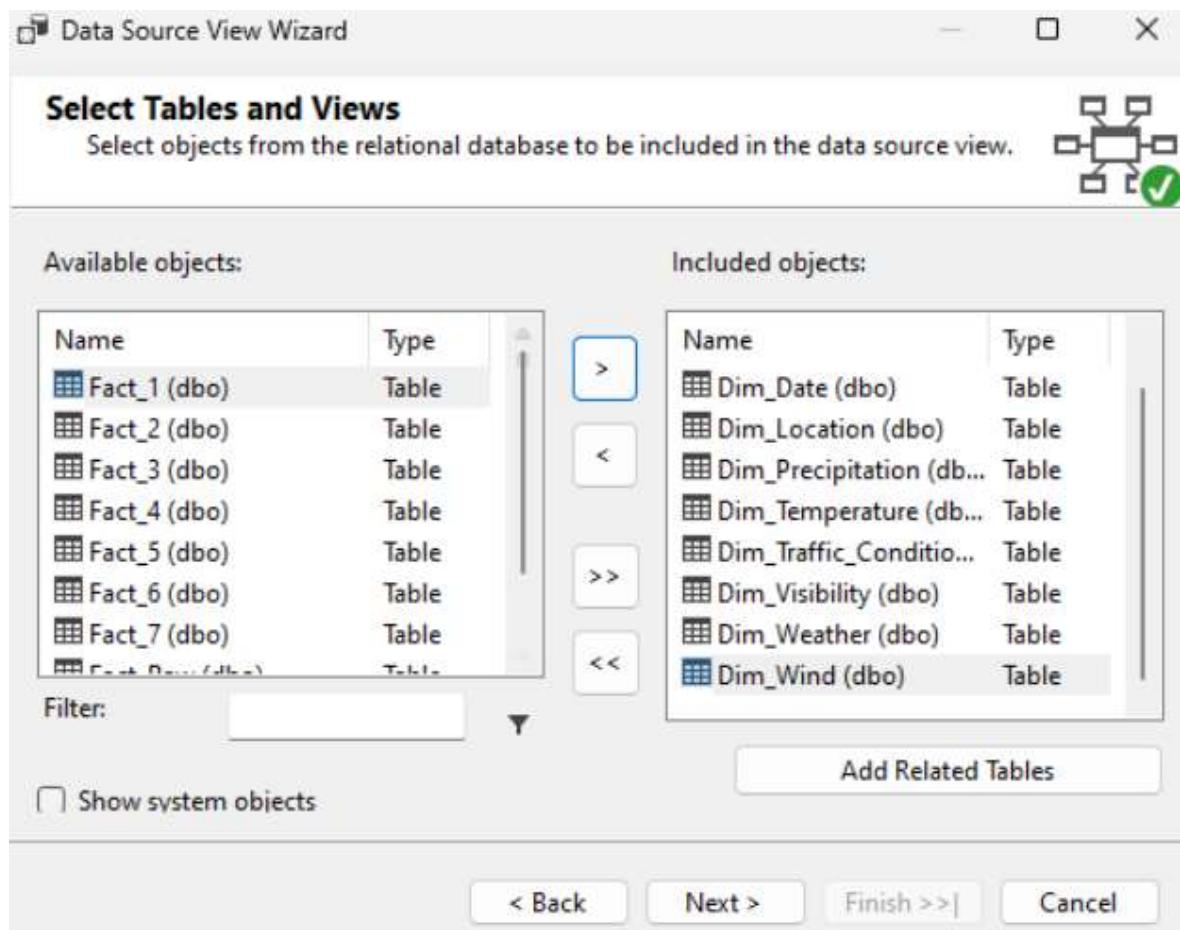
- **Bước 4:** Chọn bảng Fact, sau đó chọn nút > để thêm bảng Fact vào data source view.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



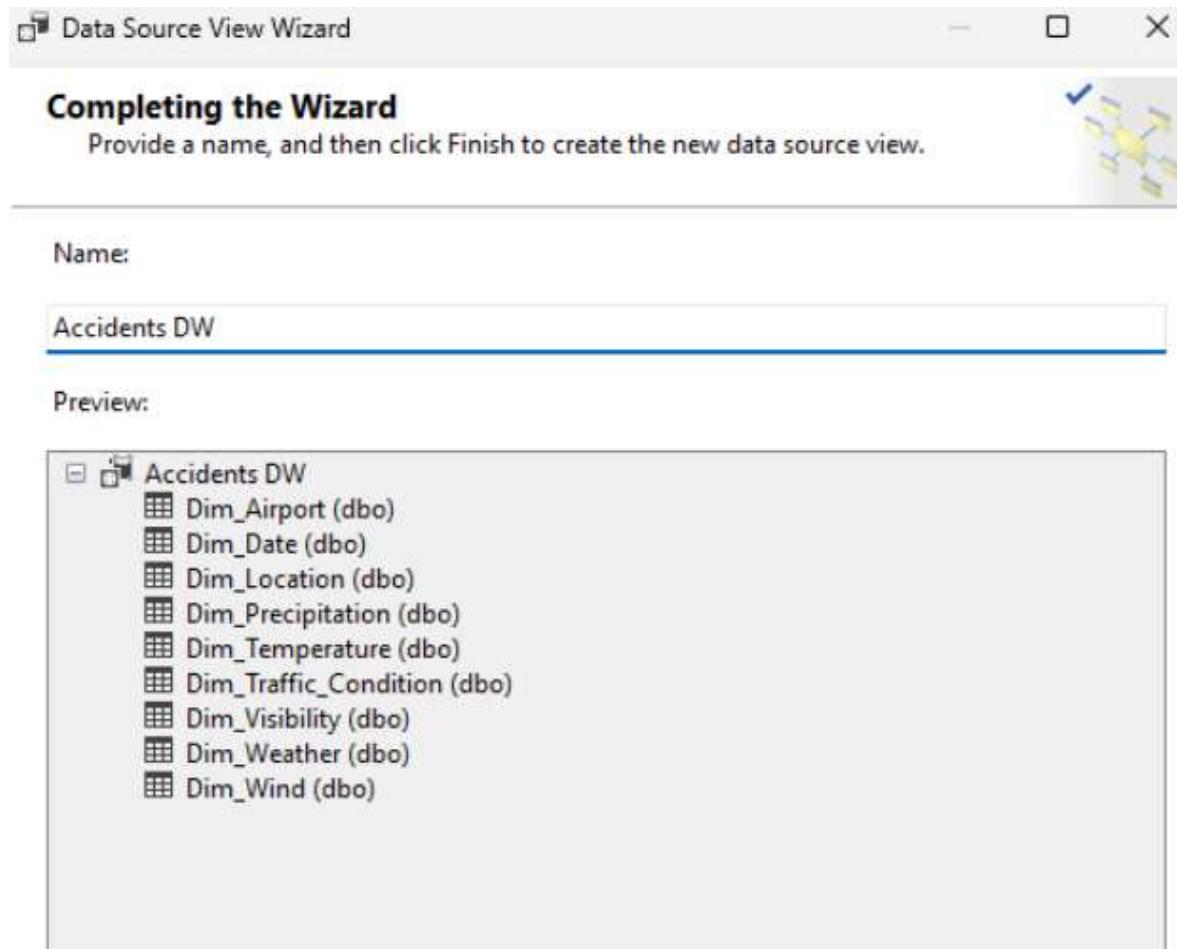
- **Bước 5:** Tiếp theo, chọn nút Add Related Tables để thêm tất cả các bảng Dim vào data source view. Sau đó chọn Next để tiếp tục.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

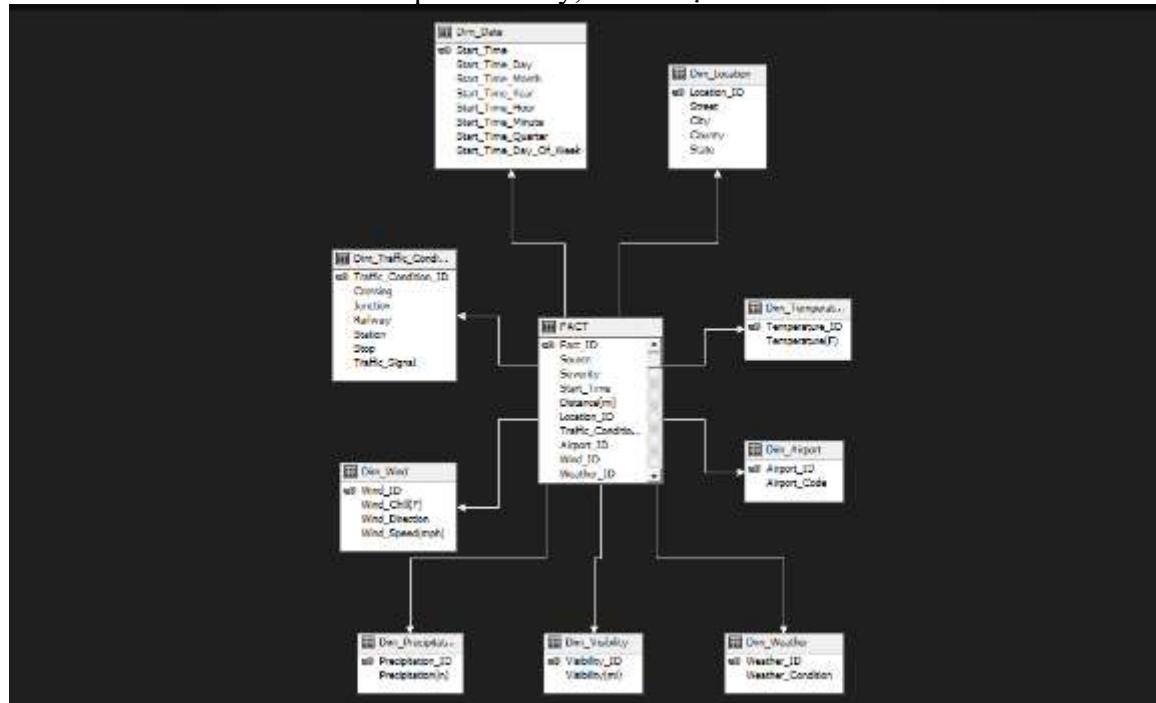


- **Bước 6:** Chọn Finish để hoàn tất quá trình xác định khung nhìn dữ liệu nguồn (Data Source View).

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

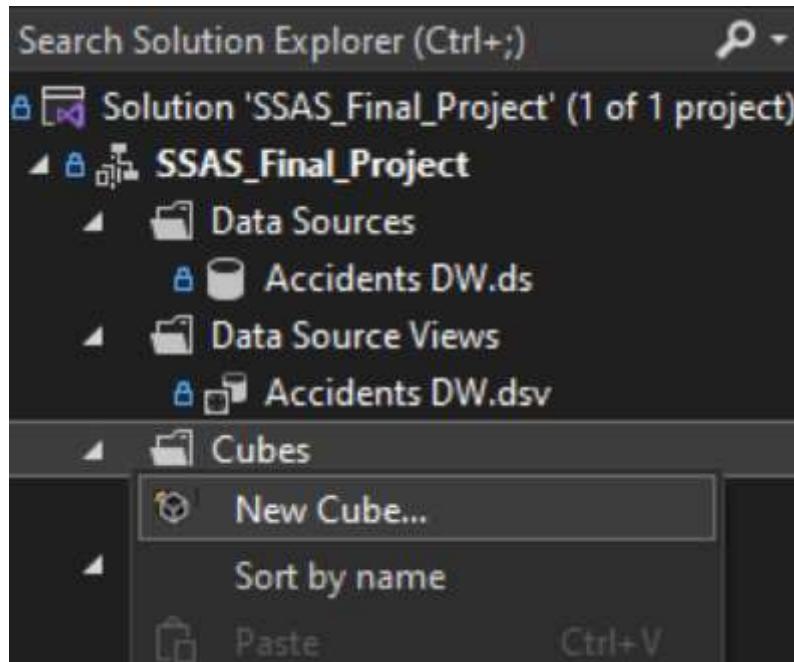


- Sau khi kết thúc quá trình này, ta sẽ được data source view như hình sau

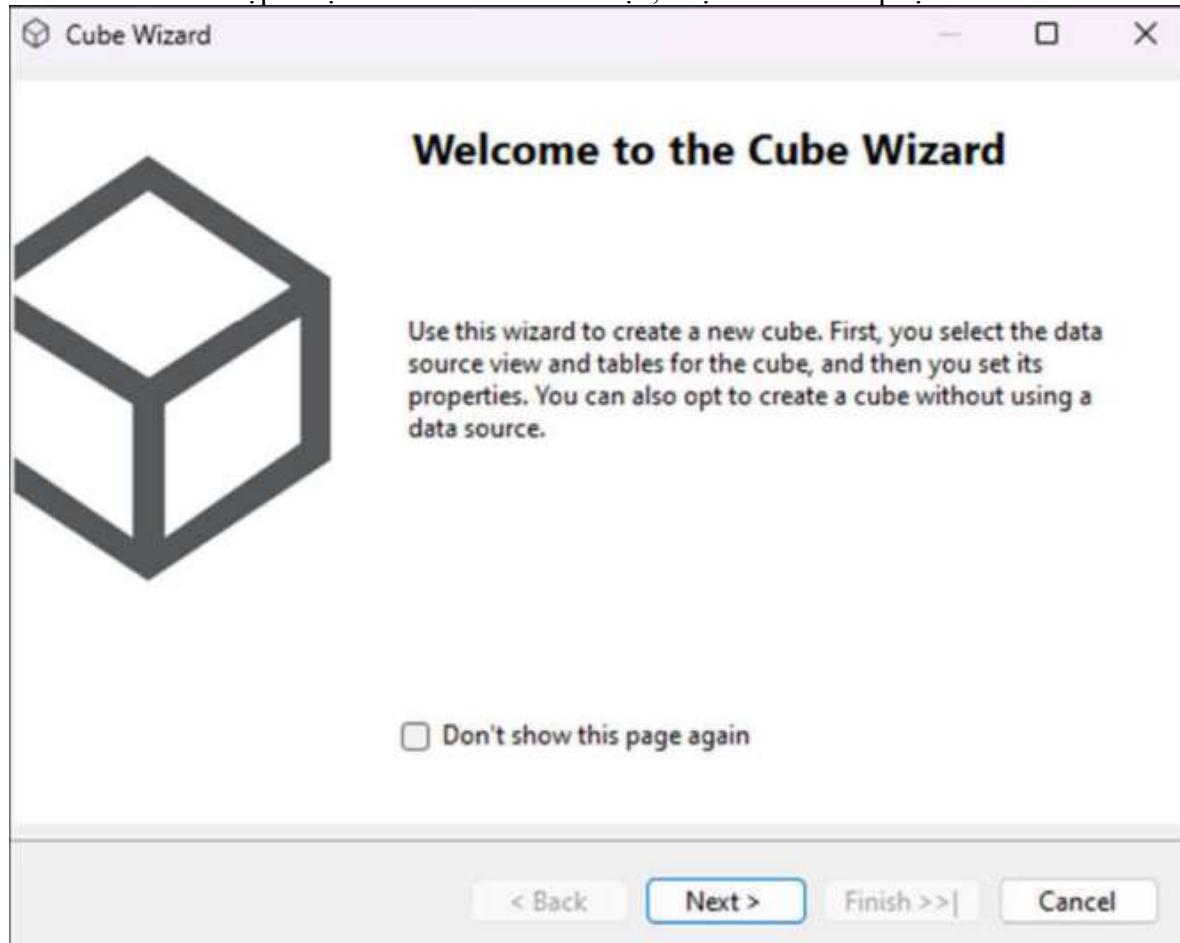


## 2. XÂY DỰNG CÁC KHỐI (CUBES) VÀ XÁC ĐỊNH CÁC ĐỘ ĐO (MEASURE)

- **Bước 1:** Tại Solution Explorer, ta click chuột phải vào thư mục Cubes và chọn New Cube.

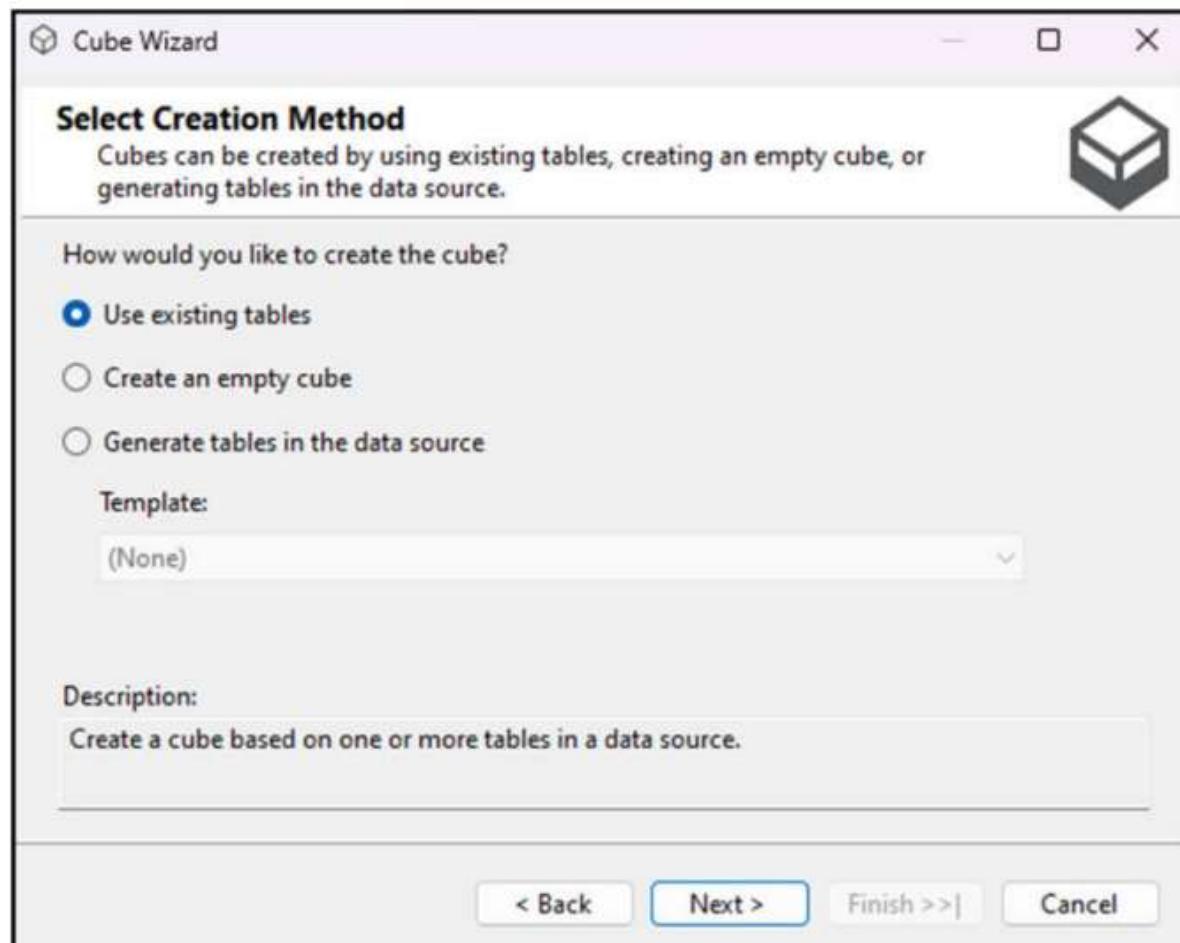


- **Bước 2:** Hộp thoại Cube Wizard xuất hiện, chọn Next để tiếp tục.



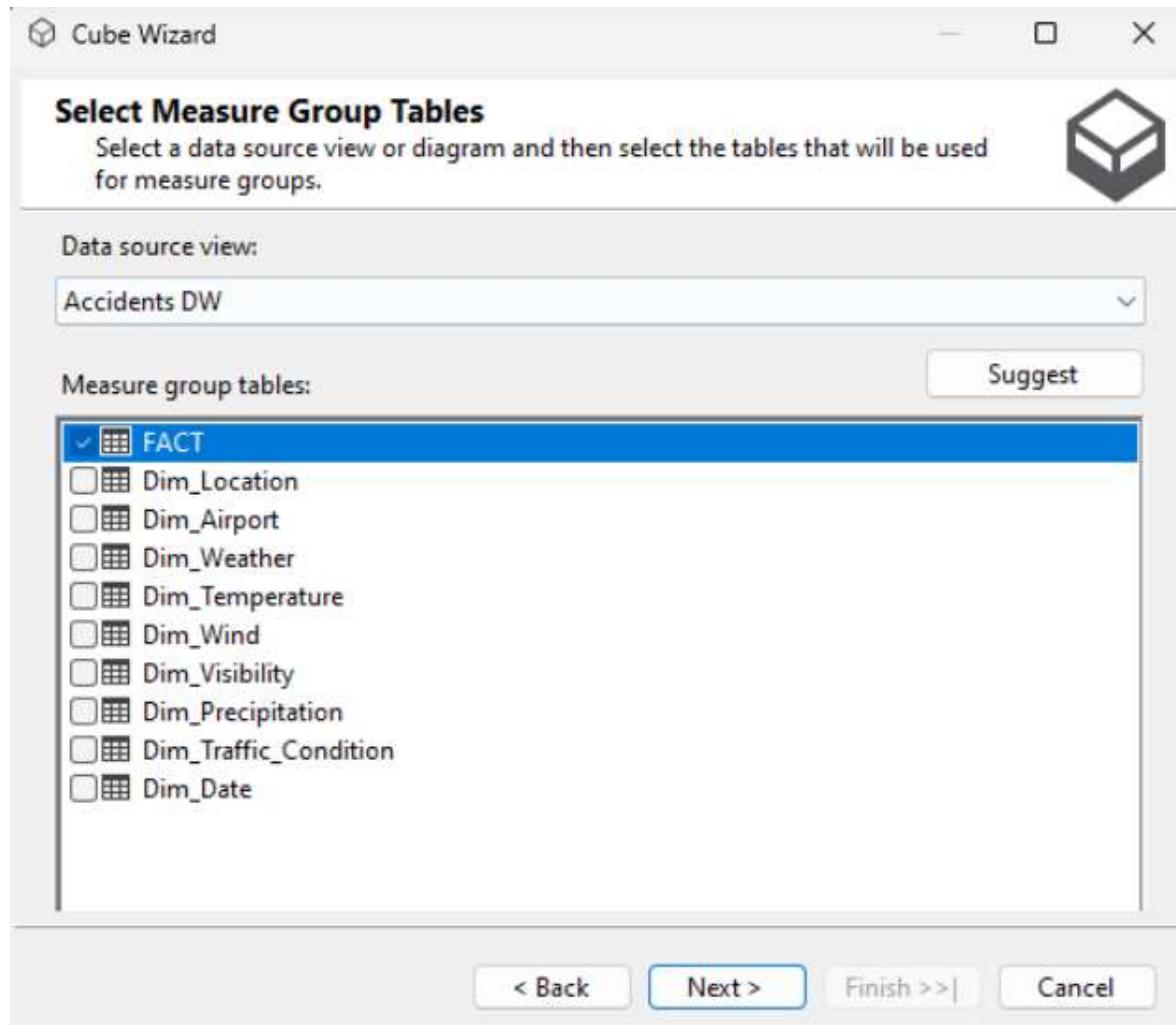
## *Đồ án xây dựng kho dữ liệu US ACCIDENTS*

- **Bước 3:** Chọn use existing tables, sau đó chọn Next để tiếp tục.



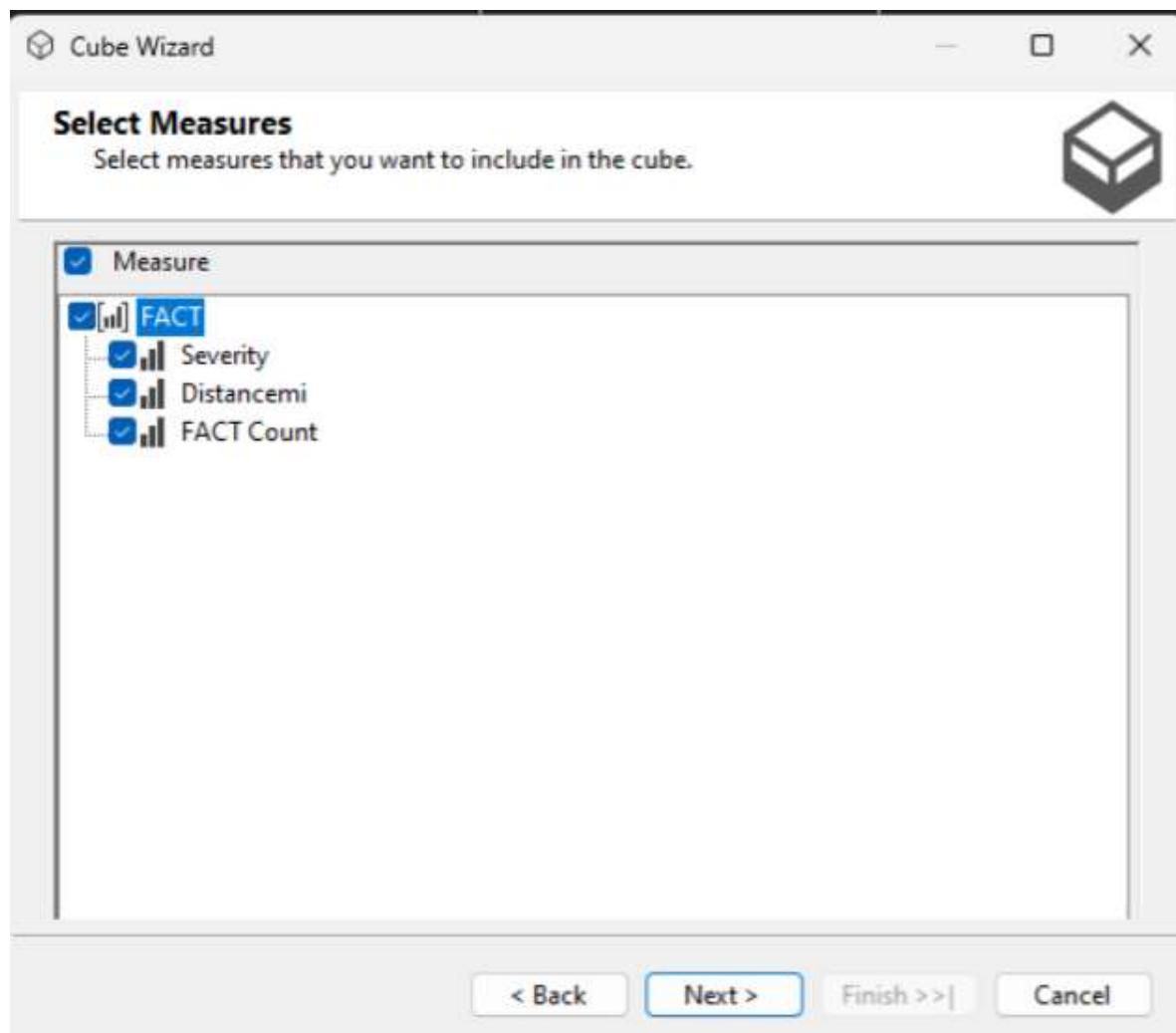
- **Bước 4:** Chọn Fact để phân chia các measure group.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



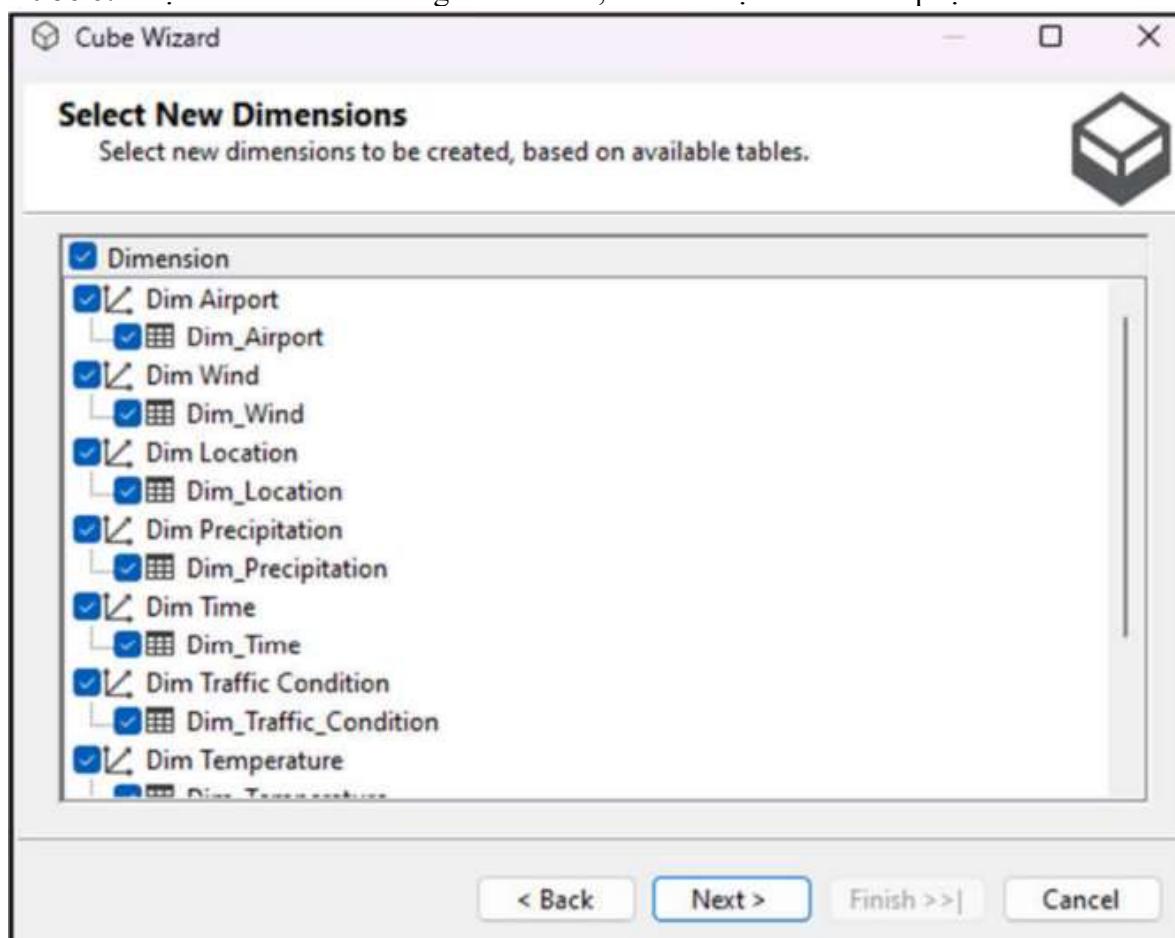
- **Bước 5:** Chọn những độ đo để xuất, sau đó chọn Next để tiếp tục.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



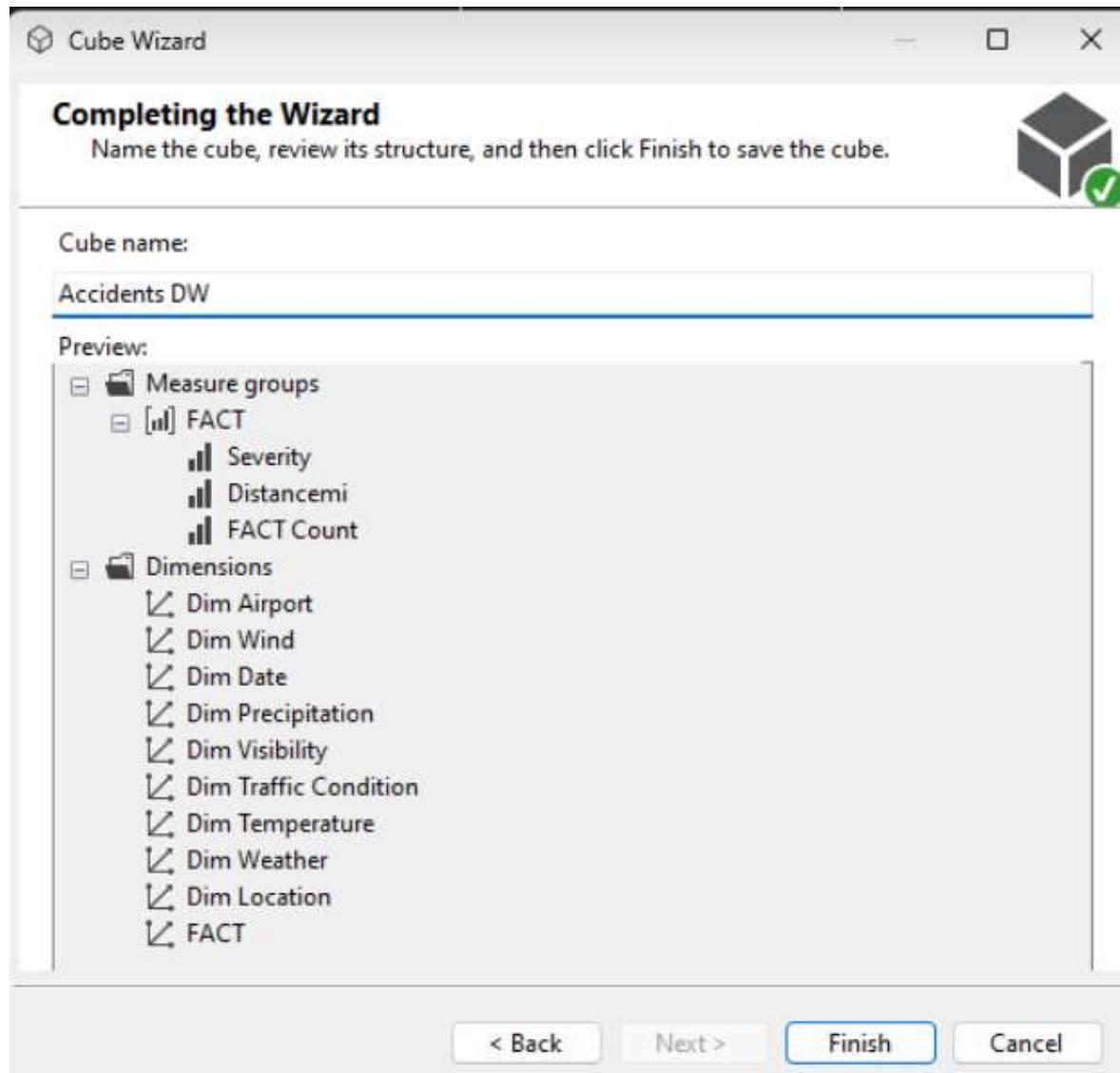
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- **Bước 6:** Chọn danh sách các bảng Dimension, sau đó chọn Next để tiếp tục.



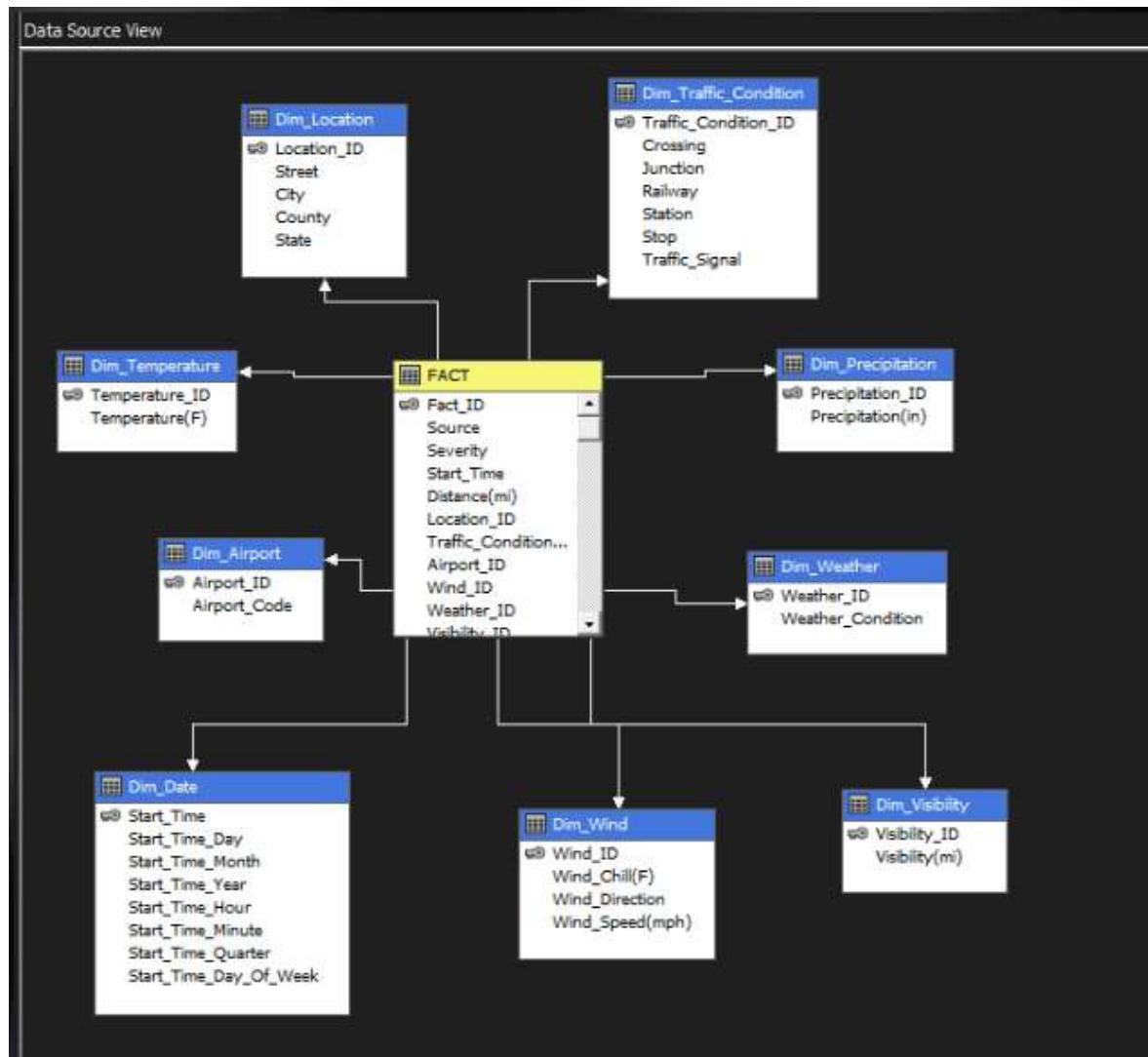
- **Bước 7:** Chọn Finish để hoàn tất quy trình xây dựng các khối (Cubes) và xác định các độ đo (Measures).

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

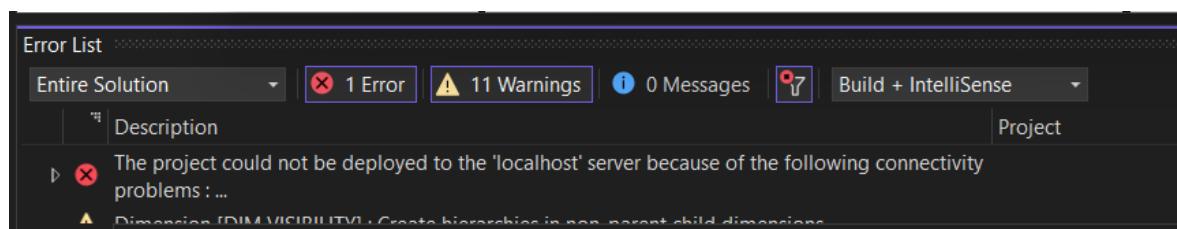


- Sau khi kết thúc quá trình này, ta sẽ được kết quả như hình sau:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

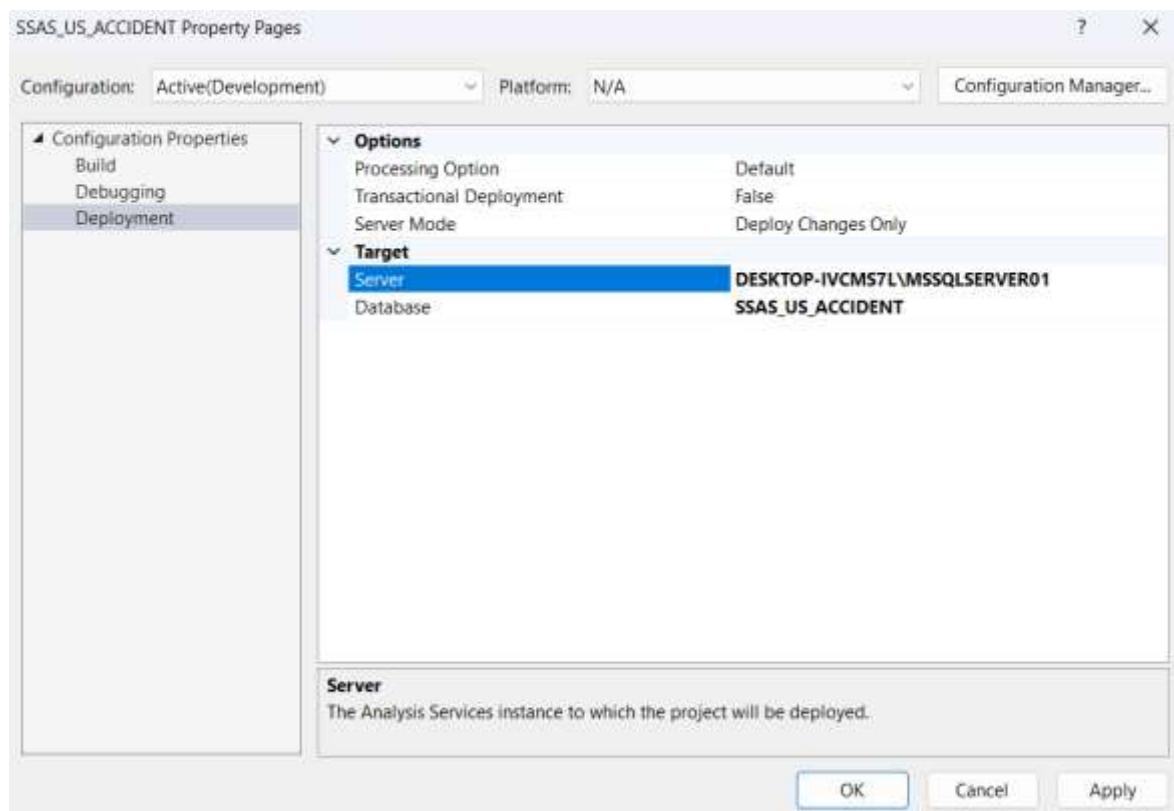


### 3. XÁC ĐỊNH CÁC CHIỀU (DEFINE A DIMENSION)

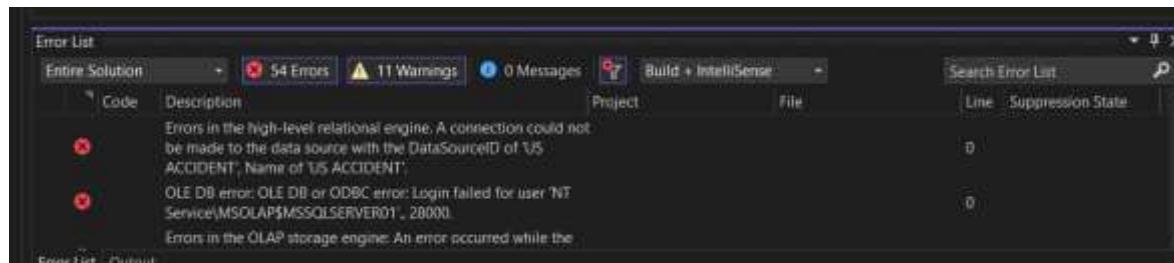


- Mở Property của dự án SSAS\_US\_Accidents và đặt Target Server thành server đã sử dụng ở các phần trước.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Sau khi đã đặt Target Server, tiến hành triển khai (deploy) dự án bằng cách bấm nút Start trên thanh công cụ. Trong quá trình triển khai dự án, có thể xảy ra lỗi như hình bên dưới:



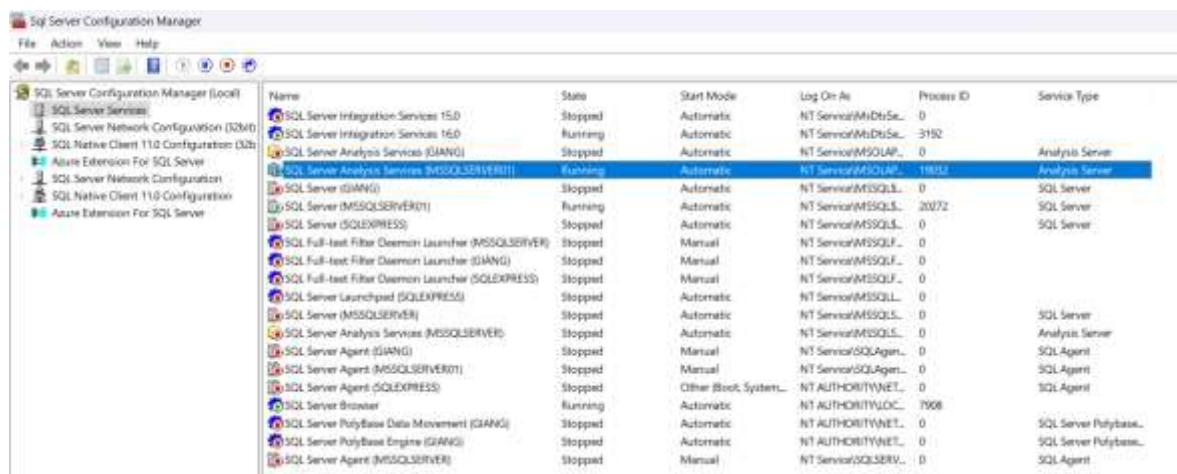
*Lỗi OLE DB error (mã lỗi: 28000) có nghĩa là Visual Studio không thể đăng nhập vào tài khoản dịch vụ (service account) của Analysis Service, hoặc tài khoản này không có quyền đọc, ghi dữ liệu trong cơ sở dữ liệu USRailAccidents. Để khắc phục lỗi này, có thể thực hiện các bước sau:*

- **Bước 1:** Mở ứng dụng SQL Server Configuration Manager.



- **Bước 2:** Click đúp chuột vào dịch vụ SQL Server Analysis Services.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

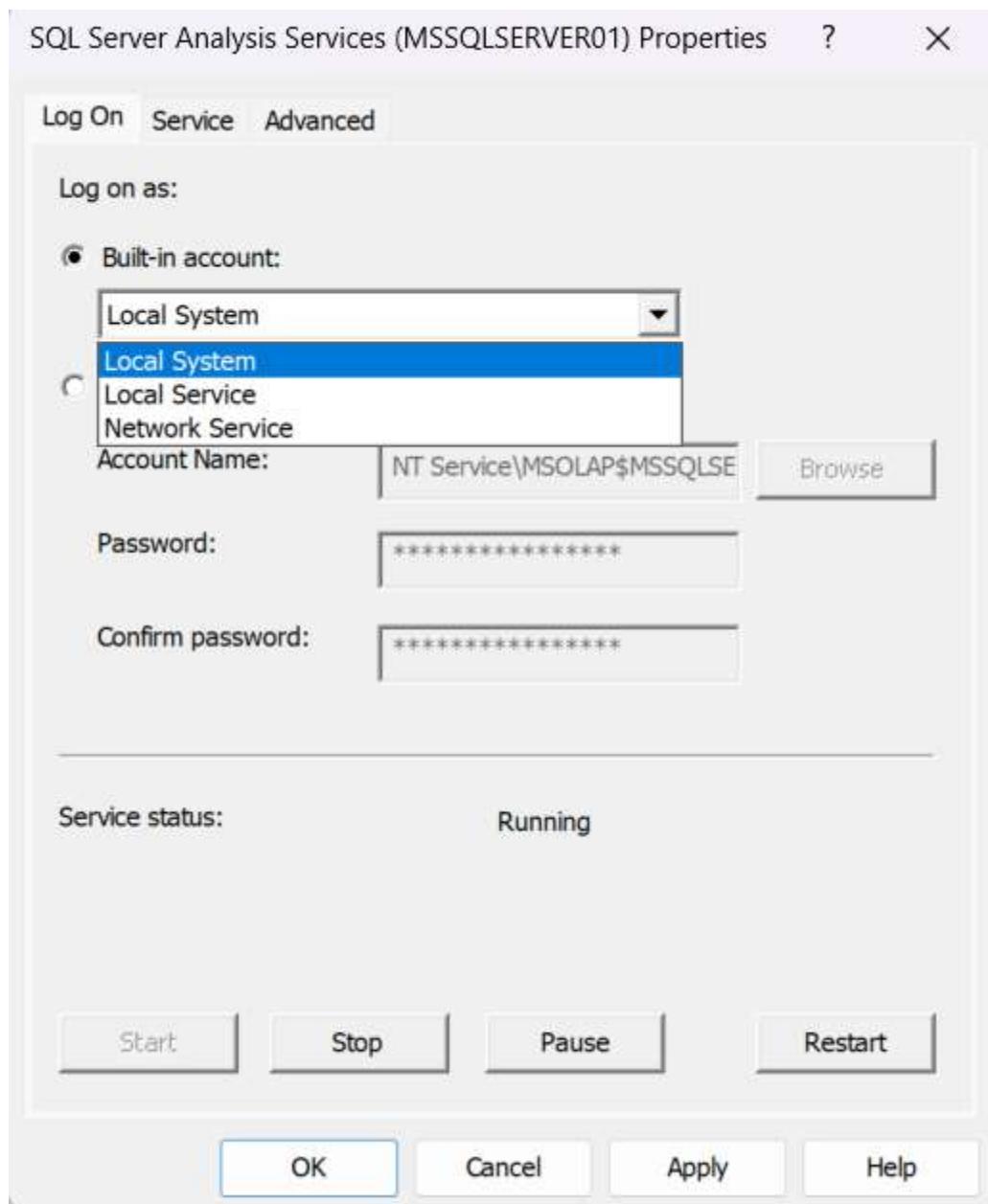


The screenshot shows the 'SQL Server Configuration Manager' window with the 'Services' tab selected. The left pane displays a tree view of configuration categories, and the right pane lists all services running on the system. The service 'SQL Server Analysis Services (MSSQLSERVER01)' is highlighted in blue.

Name	Status	Start Mode	Log On As	Process ID	Service Type
SQL Server Integration Services	Stopped	Automatic	NT Service\MSDTS...	0	
SQL Server Network Configuration (32bit)	Running	Automatic	NT Service\MSDTCSe...	3192	
SQL Native Client 11.0 Configuration (32bit)	Stopped	Automatic	NT Service\MSOAC...	0	
Azure Extension For SQL Server	Running	Automatic	NT Service\AZURE...	19832	Analysis Server
SQL Server Network Configuration	Running	Automatic	NT Service\MSQL...	2072	SQL Server
SQL Native Client 11.0 Configuration	Stopped	Automatic	NT Service\MSOAC...	0	SQL Server
Azure Extension For SQL Server	Running	Automatic	NT Service\AZURE...	19832	Analysis Server
SQL Server Analysis Services (MSSQLSERVER01)	Running	Automatic	NT Service\MSOLAP...	19832	Analysis Server
SQL Server (MSSQLSERVER01)	Stopped	Automatic	NT Service\MSQL...	0	SQL Server
SQL Server (SQLEXPRESS)	Stopped	Automatic	NT Service\MSQL...	0	SQL Server
SQL Full-text Filter Daemon Launcher (MSSQLSERVER)	Stopped	Manual	NT Service\MSQOLF...	0	
SQL Full-text Filter Daemon Launcher (GANG)	Stopped	Manual	NT Service\MSQOLF...	0	
SQL Full-text Filter Daemon Launcher (SQLEXPRESS)	Stopped	Manual	NT Service\MSQOLF...	0	
SQL Server Launchpad (SQLEXPRESS)	Stopped	Automatic	NT Service\MSQL...	0	
SQL Server (MSSQLSERVER)	Stopped	Automatic	NT Service\MSQL...	0	SQL Server
SQL Server Analysis Services (MSSQLSERVER)	Stopped	Automatic	NT Service\MSOLAP...	0	Analysis Server
SQL Server Agent (GANG)	Stopped	Manual	NT Service\SQLAgen...	0	SQL Agent
SQL Server Agent (MSSQLSERVER01)	Stopped	Manual	NT Service\SQLAgen...	0	SQL Agent
SQL Server Agent (SQLEXPRESS)	Stopped	Other (Boot, System...)	NT AUTHORITY\NET...	0	SQL Agent
SQL Server Browser	Running	Automatic	NT AUTHORITY\NET...	7908	
SQL Server PolyBase Data Movement (GANG)	Stopped	Automatic	NT AUTHORITY\NET...	0	SQL Server Polybase...
SQL Server PolyBase Engine (GANG)	Stopped	Automatic	NT AUTHORITY\NET...	0	SQL Server Polybase...
SQL Server Agent (MSSQLSERVER)	Stopped	Manual	NT Service\SQLSERV...	0	SQL Agent

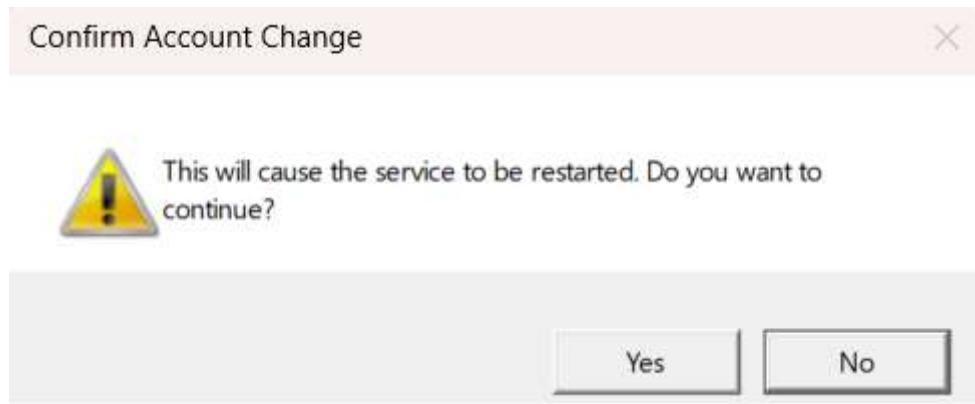
- **Bước 3:** Hộp thoại Properties xuất hiện. Tại đây, chọn tài khoản đăng nhập (Log on as) là **Built-in account -> Local System**.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 4:** Click OK -> Yes để khởi động lại SQL Server Analysis Services và áp dụng các thay đổi.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

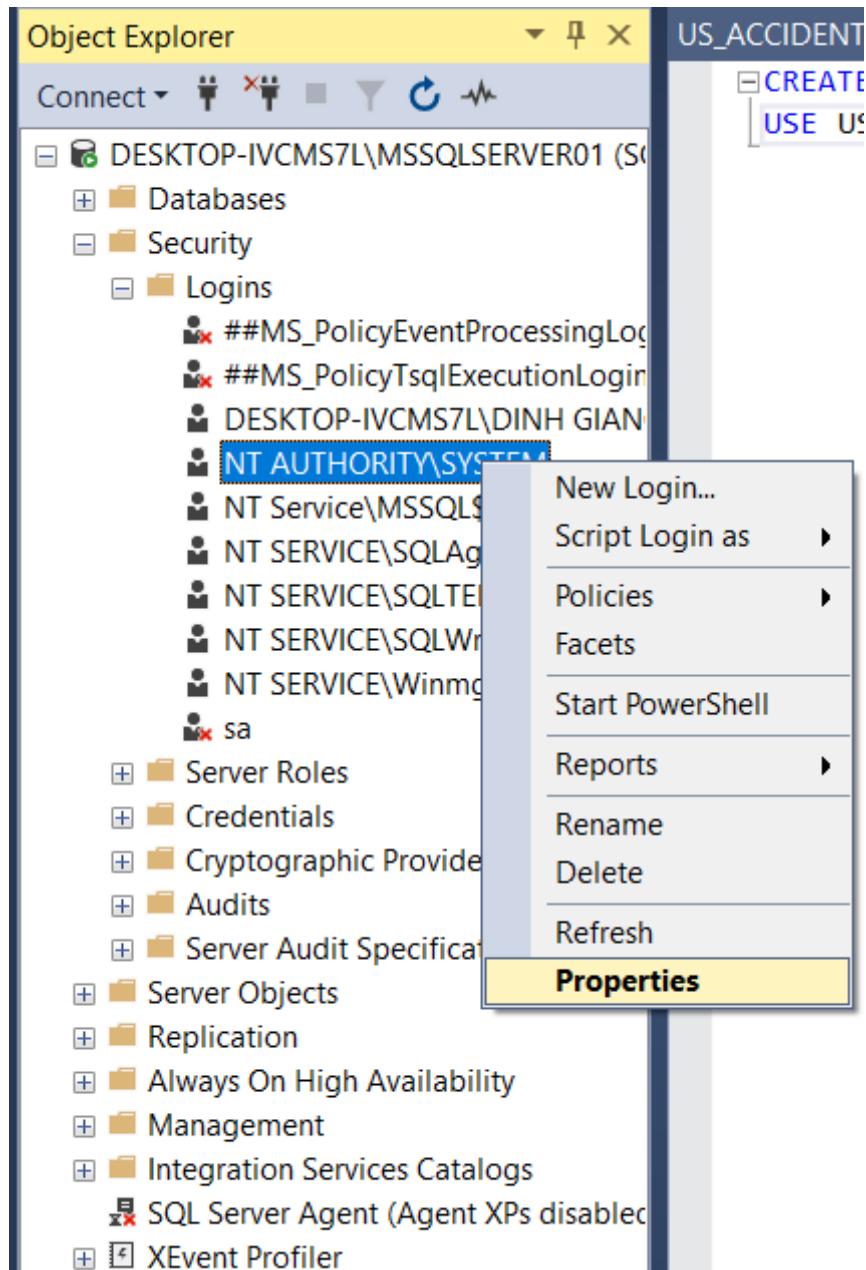


- **Bước 5:** Sau khi dịch vụ SSAS khởi động lại, ta thấy tài khoản Log On của SSAS đã thay đổi thành **LocalSystem**.

SQL Server Configuration Manager				
File Action View Help				
Name	State	Start Mode	Log On As	
SQL Server Integration Services 15.0	Stopped	Automatic	NT Service\MSDTS...	
SQL Server Integration Services 16.0	Running	Automatic	NT Service\MSDTS...	
SQL Server Analysis Services (GIANG)	Stopped	Automatic	NT Service\MSOLAP...	
SQL Server Analysis Services (MSSQLSERVER01)	Running	Automatic	LocalSystem	
SQL Server (GIANG)	Stopped	Automatic	NT Service\MSSQLS...	
SQL Server (MSSQLSERVER01)	Running	Automatic	NT Service\MSSQLS...	
SQL Server Agent (MSSQLSERVER01)	Running	Automatic	NT Service\MSAS...	

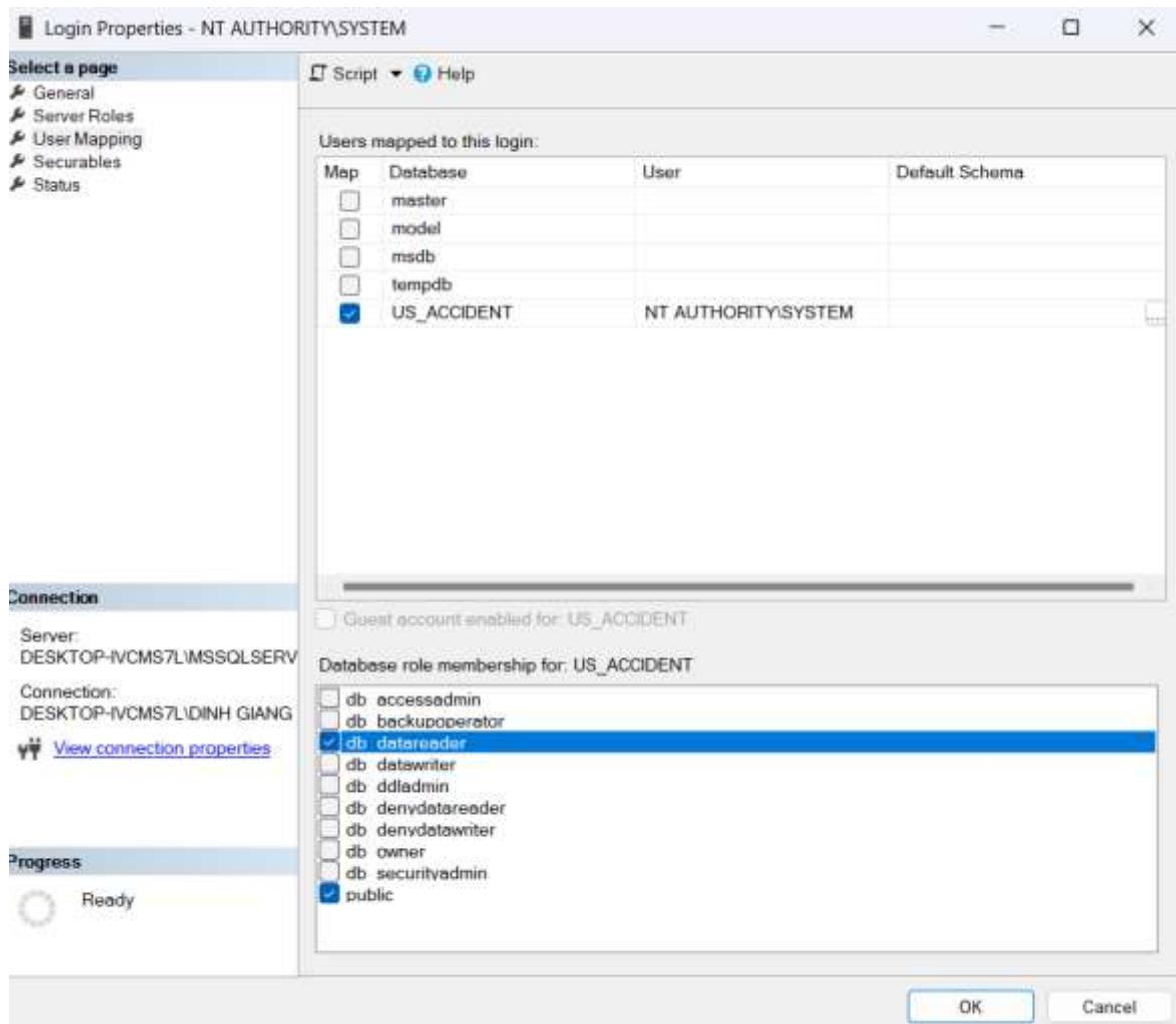
- **Bước 6:** Mở Microsoft SQL Server Management Studio (MSSMS) và kết nối với Database Engine **MSI|MSSQLSERVER1** (server dùng để lưu cơ sở dữ liệu USRailAccidents). Click chuột phải vào tài khoản **NT AUTHORITY\SYSTEM** trong Security\Logins và mở Properties

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 7:** Tại trang User Mapping, chọn database USRailAccidents trong danh sách, sau đó chọn vai trò **db datareader** tại danh sách các vai trò truy cập cơ sở dữ liệu bên dưới. Click OK để áp dụng thay đổi.

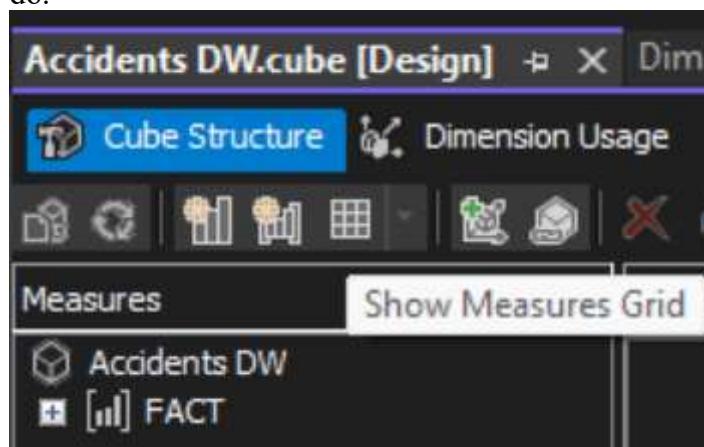
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Sau khi hoàn thành các bước trên, bấm Start để triển khai lại dự án SSAS. Quá trình triển khai đã diễn ra thành công.

## 4. XÁC ĐỊNH CÁC ĐỘ ĐO (MEASURES)

- Bước 1: Tại khôi vừa tạo, chọn Show Measures Grid để hiện thị chi tiết các độ đo.



## Đồ án xây dựng kho dữ liệu US ACCIDENTS

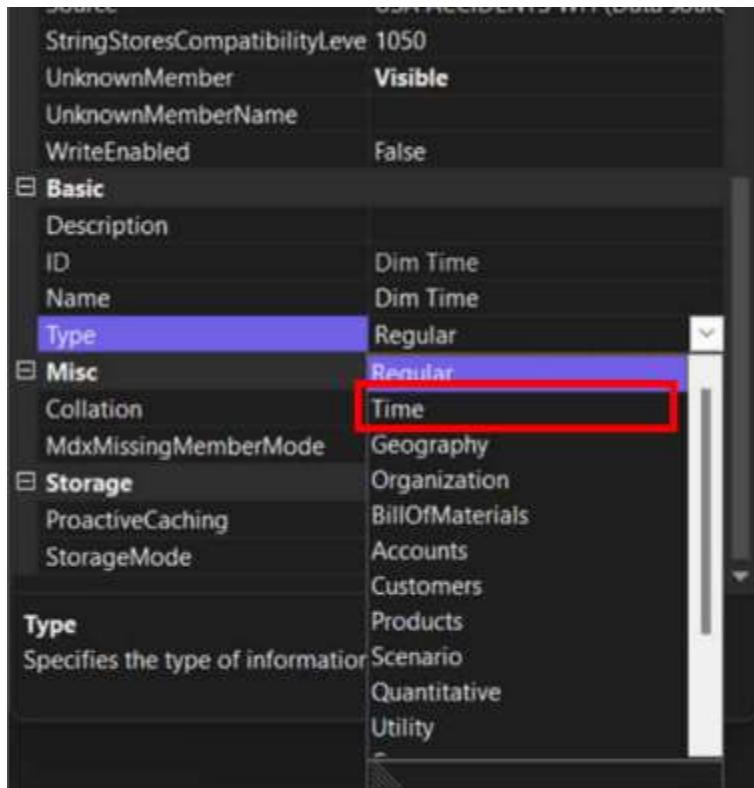
- Chi tiết các độ đo sẽ hiển thị dưới dạng bảng, dễ dàng để tương tác.
- Bước 2:** Ta đổi tên và thuộc tính các độ đo hiện tại theo các hàm tổng hợp (aggregation) như sau:
  - Severity (mức độ nghiêm trọng của vụ tai nạn) đổi thành Average Severity (mức độ nghiêm trọng trung bình của vụ tai nạn), sau đó đổi Aggregation từ Sum sang Average.
  - Distance (chiều dài của đoạn đường bị ảnh hưởng bởi vụ tai nạn) đổi thành Sum Distance (tổng chiều dài của các đoạn đường bị ảnh hưởng bởi vụ tai nạn).
  - Fact Count (Số lượng vụ tai nạn) đổi thành Fact Count (Số lượng vụ tai nạn xảy ra tại Mỹ) và giữ nguyên Aggregation là Count.

Name	Measure Group	Data Type	Aggregation
Average Severity	FACT	Double	AverageOfChild...
Sum Distance	FACT	Double	Sum
FACT Count	FACT	Integer	Count
Add new measure...			

- Sau khi đổi tên và thuộc tính các độ đo ban đầu. Một thông báo xuất hiện yêu cầu ta phải có một time dimension.

- Bước 3:** Mở Dim Time.dim. Tại cửa sổ Properties, ta đổi kiểu bảng từ Regular sang Time.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Quá trình hoàn tất, ta có được các độ đo: Average Severity, Sum Distance và Fact Count.

The screenshot shows the 'Accidents DW.cube [Design]' cube structure. In the 'Measures' section, there are three defined measures:

Name	Measure Group	Data Type	Aggregation
Average Severity	FACT	Double	AverageOfChild
Sum Distance	FACT	Double	Sum
FACT Count	FACT	Integer	Count

## 5. PHÂN CẤP TRONG BẢNG CHIỀU

### 5.1. PHÂN CẤP TRONG BẢNG Dim\_Time

- Bước 1: Kéo những thuộc tính cần phân cấp qua cửa sổ Hierarchies

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

The screenshot shows the Dimension Structure tool interface. The top bar displays three tabs: "Dim Time.dim [Design]" (selected), "USA ACCIDENT...lube [Design]\*", and "Dim Temperature.dim [Design]". Below the tabs are four buttons: "Dimension Struct...", "Attribute Relationships", "Translations", and "Browser". The main area is divided into two panes: "Attributes" on the left and "Hierarchies" on the right.

**Attributes pane:**

- Grouped under "Dim Time":
  - Start Time Date
  - Start Time Date Of Week
  - Start Time Day
  - Start Time Hour
  - Start Time Minute
  - Start Time Month
  - Start Time Quater
  - Start Time Year

**Hierarchies pane:**

- Grouped under "Hierarchy":
  - Start Time Year
  - Start Time Day
  - Start Time Quater
  - Start Time Minute
  - Start Time Hour
  - Start Time Month
  - <new level>

A tooltip on the right side of the Hierarchies pane says: "To create a new hierarchy, drag an attribute here."

- **Bước 2:** Sắp xếp lại các thuộc tính phân cấp theo thứ tự: Start Time Year -> Start Time Quarter -> Start Time Month -> Start Time Day -> Start Time Hour -> Start Time Minute và đổi tên Hierarchy thành Accidents Time.

The screenshot shows the Dimension Structure tool interface with the same layout as the previous one. The "Attributes" pane remains the same. In the "Hierarchies" pane, the hierarchy has been renamed to "Accidents Time".

- Grouped under "Accidents Time":
  - Start Time Year
  - Start Time Quater
  - Start Time Month
  - Start Time Day
  - Start Time Hour
  - Start Time Minute
  - <new level>

A tooltip on the right side of the Hierarchies pane says: "To create a new hierarchy, drag an attribute here."

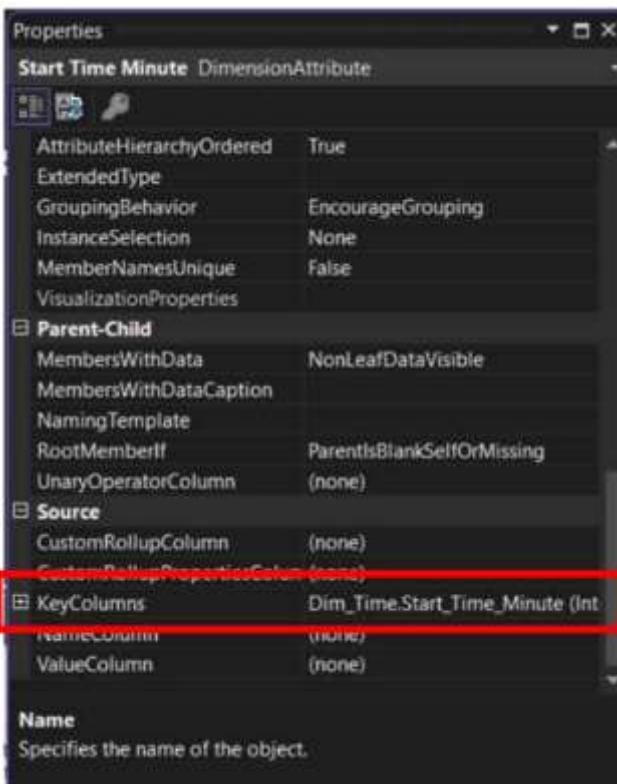
- **Bước 3:** Tại panel Attribute Relationships, tạo mối quan hệ như sau



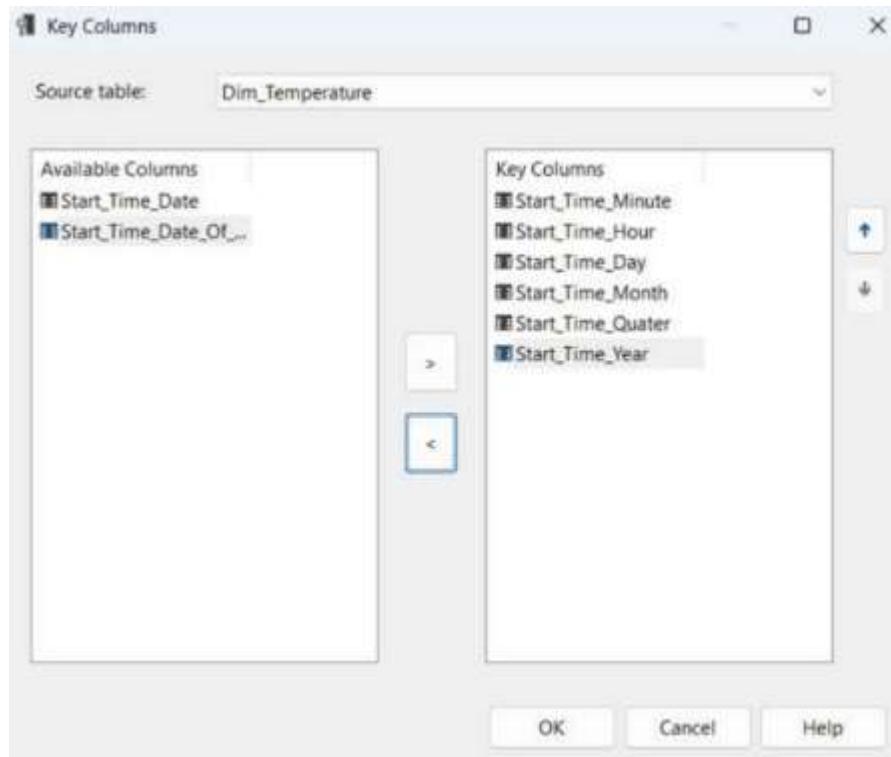
- **Bước 4:** Chính khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính Start Time Minute. Vì thuộc tính Start Time Minute là thuộc tính cấp nhỏ nhất sẽ lấy khóa cột gồm chính nó và những thuộc tính cấp cao hơn.

- Tại cửa sổ Properties của thuộc tính Start Time Minute, chọn KeyColumns.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

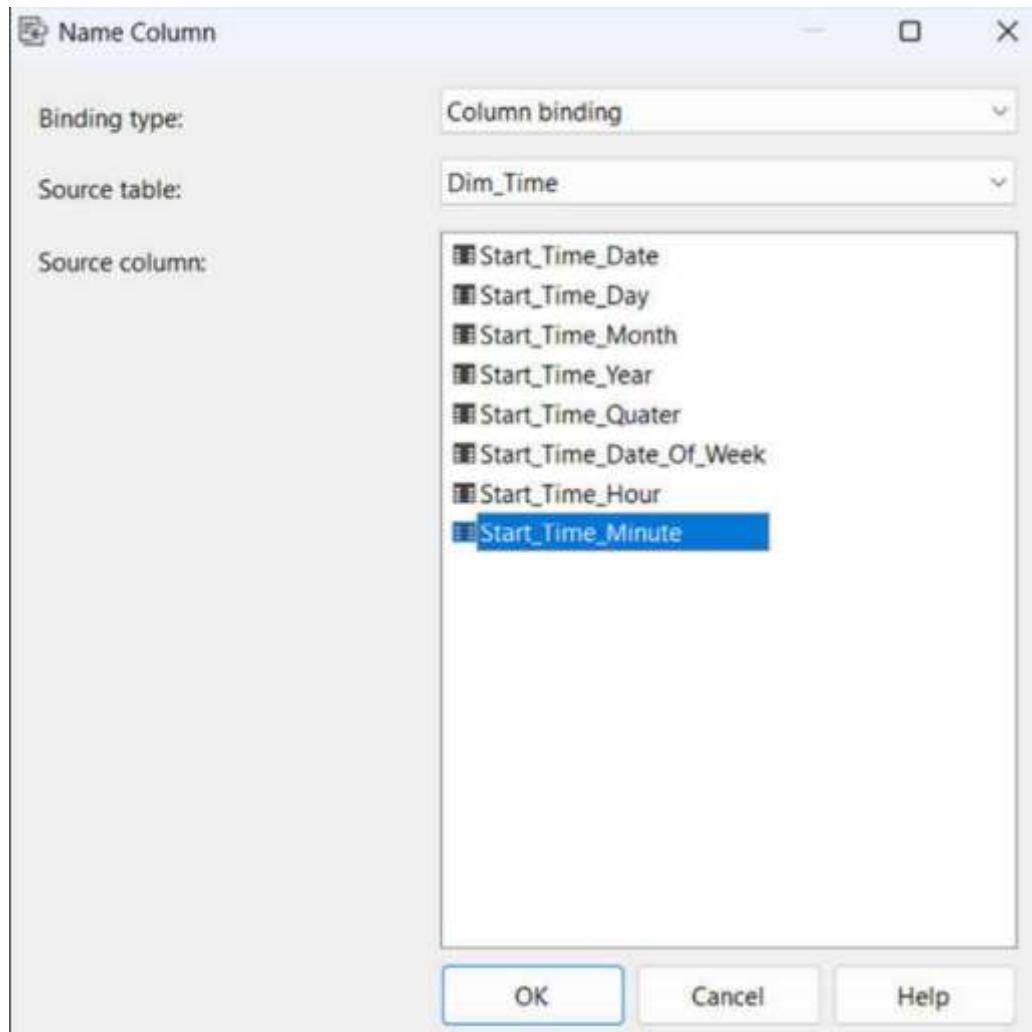


- Thêm các thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để hoàn tất.



- Tại cửa sổ Properties của thuộc tính Start Time Minute, ta chọn Name Column và chọn tên thuộc tính là Start Time Minute.

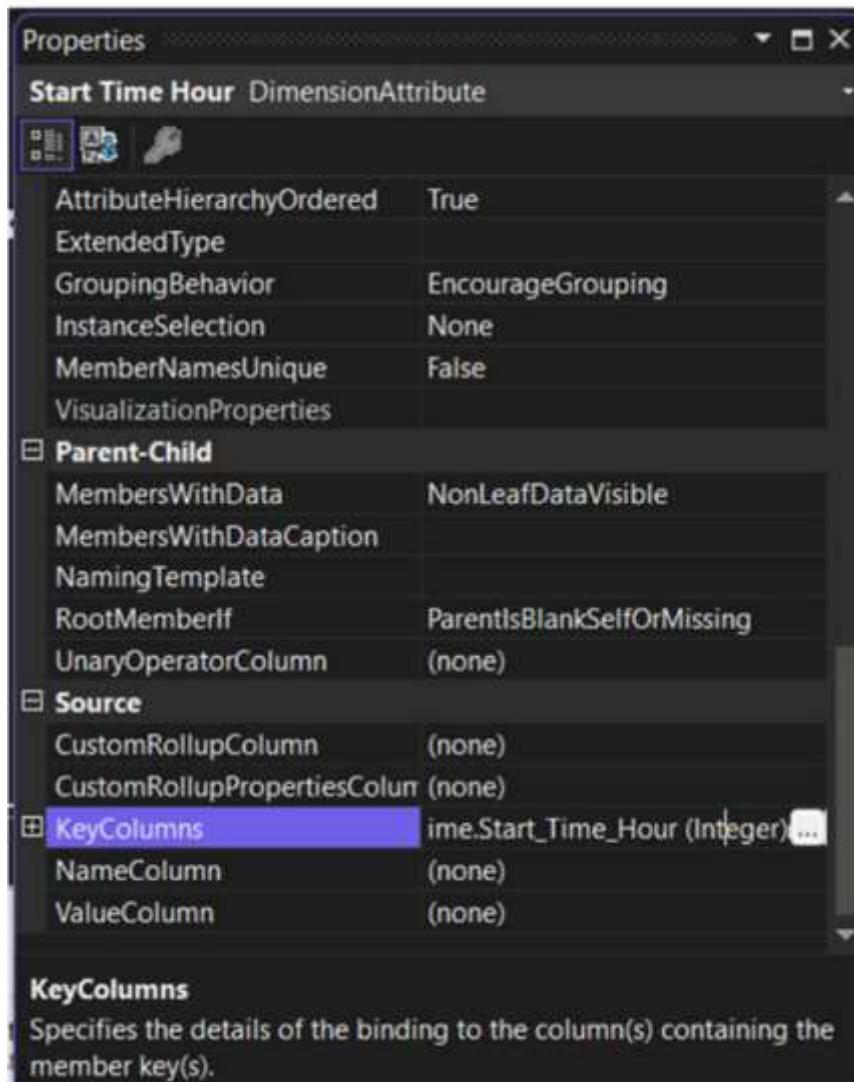
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 5:** Chỉnh khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính Start Time Hour. Vì thuộc tính Start Time Hour là thuộc tính cấp nhỏ hơn Start Time Day, Start Time Month, Start Time Quarter, Start Time Year nên sẽ lấy khóa cột gồm chính nó và những thuộc tính cấp cao hơn.

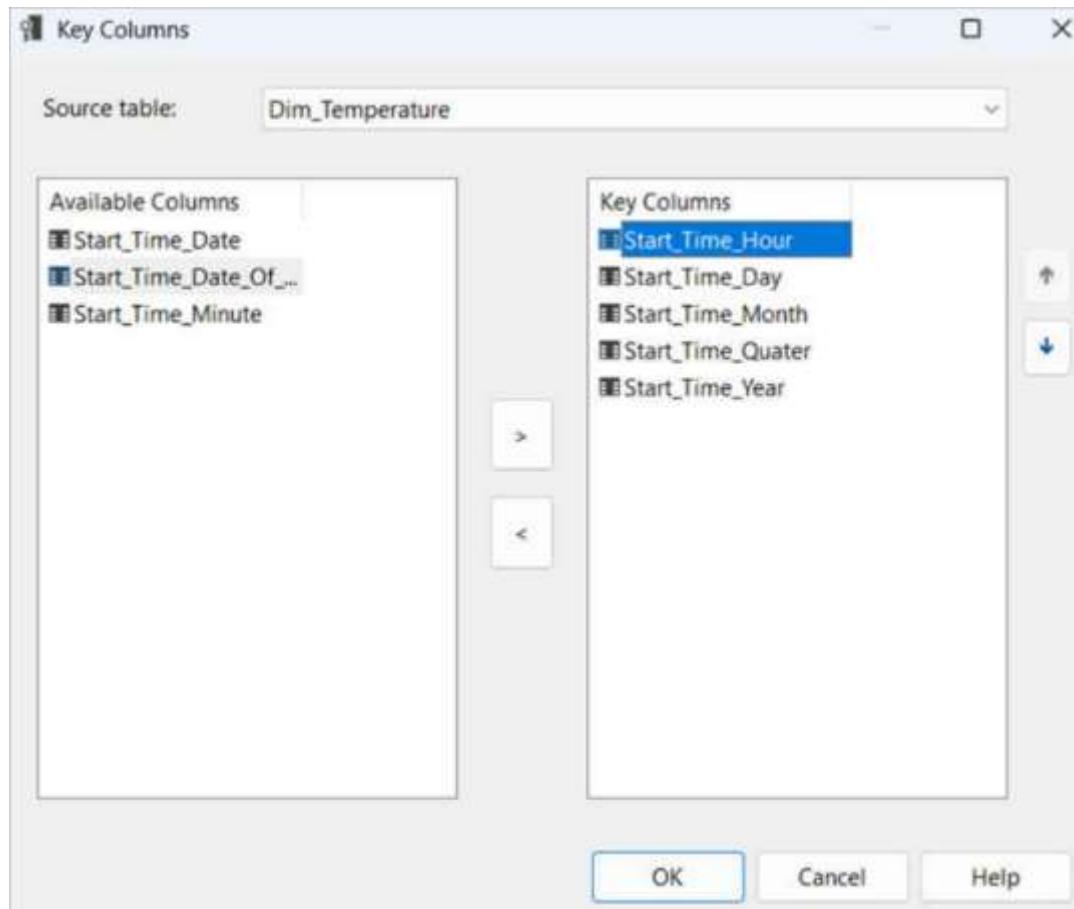
- Tại cửa sổ Properties của thuộc tính Start Time Hour, chọn KeyColumns.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

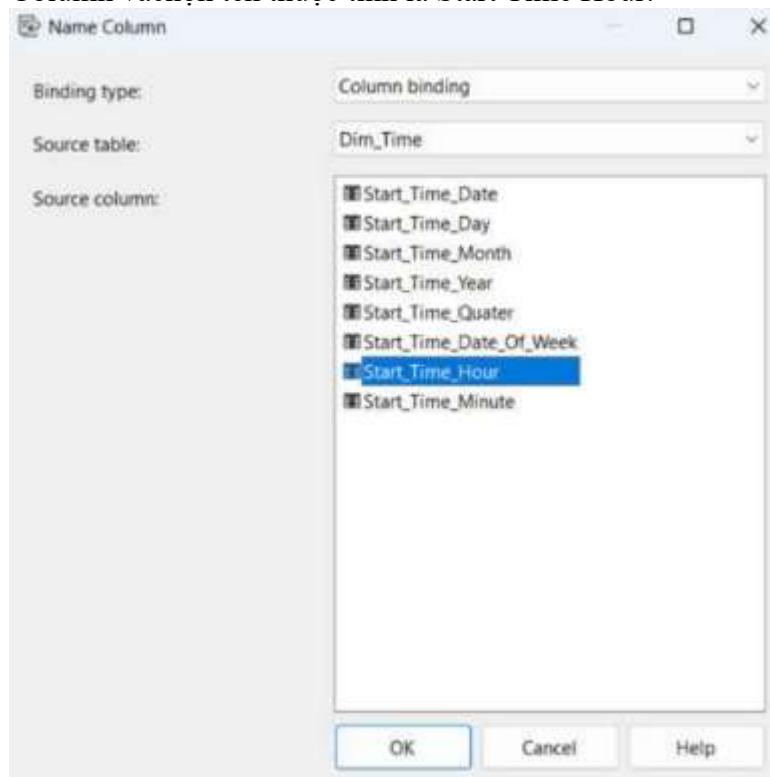


- Thêm các những thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để hoàn tất.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Tại cửa sổ Properties của thuộc tính Start Time Hour, ta chọn Name Column và chọn tên thuộc tính là Start Time Hour.

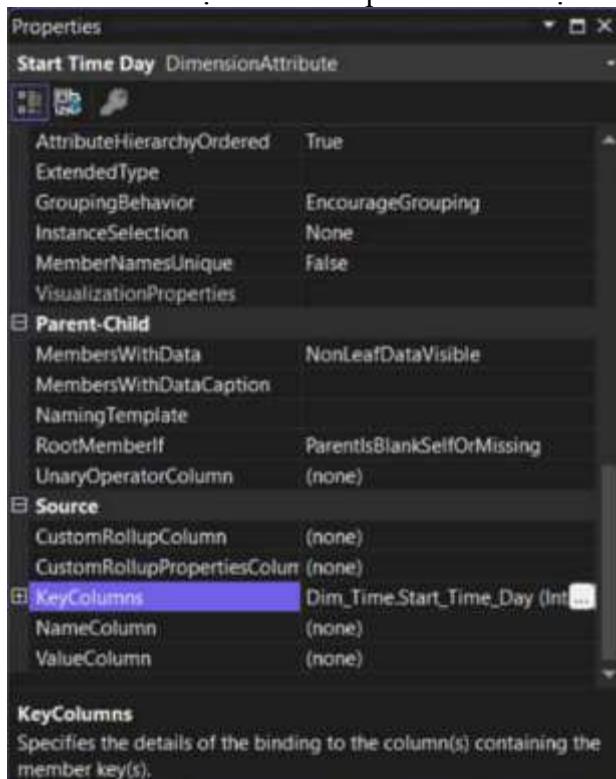


- Bước 6: Chính khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

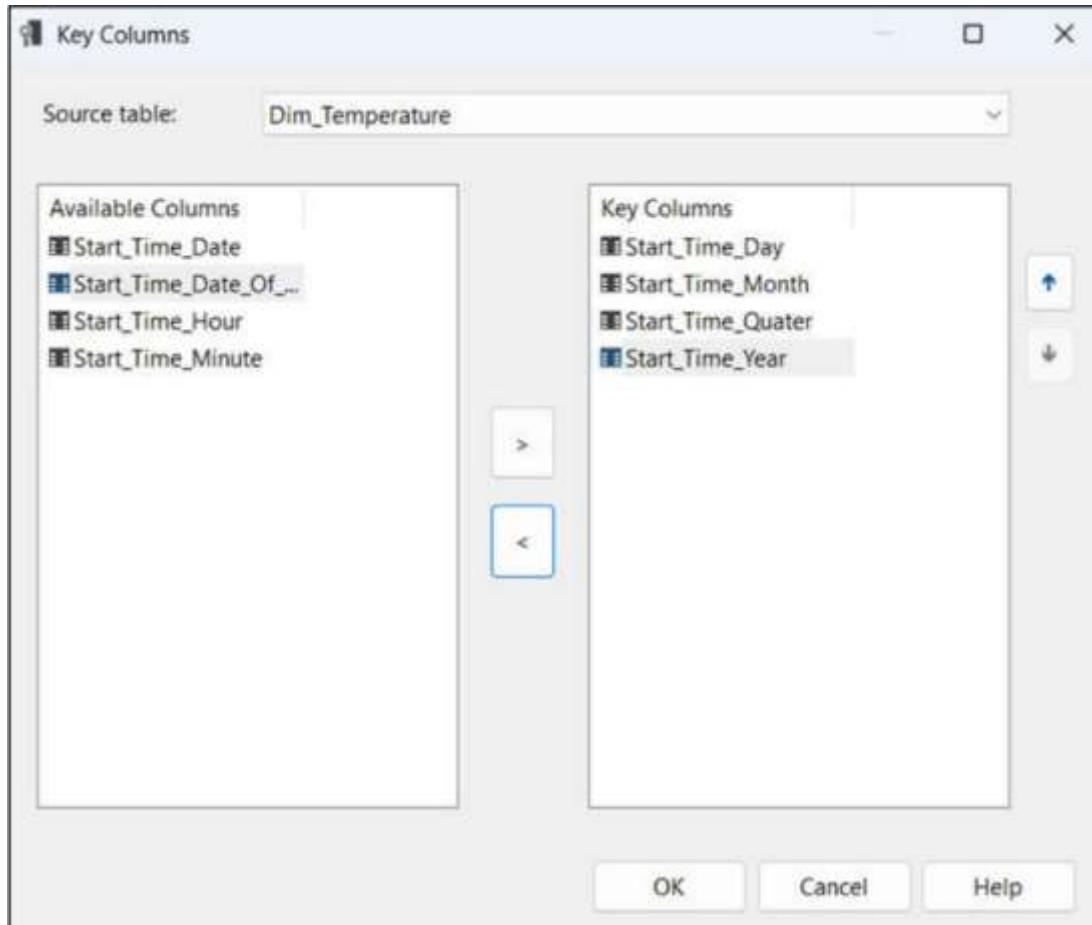
Start Time Day. Vì thuộc tính Start Time Day là thuộc tính cấp nhỏ hơn Start Time Month, Start Time Quarter, Start Time Year nên sẽ lấy khóa cột gồm chính nó và những thuộc tính cấp cao hơn.

- Tại cửa sổ Properties của thuộc tính Start Time Day, chọn KeyColumns.



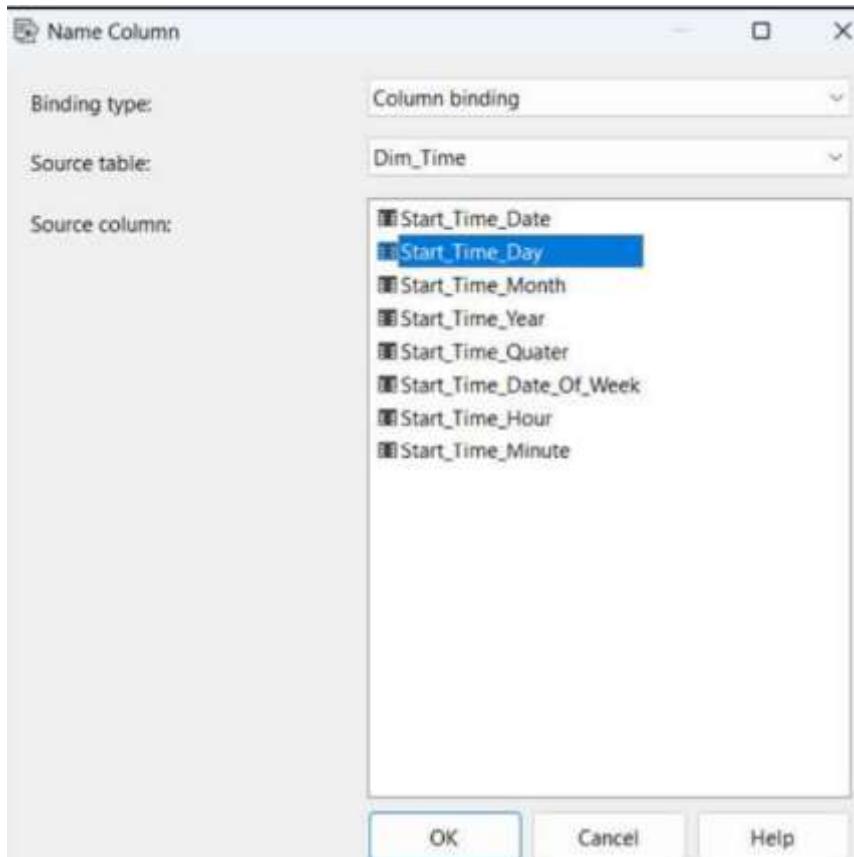
- Thêm các những thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để hoàn tất.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Tại cửa sổ Properties của thuộc tính Start Time Day, ta chọn Name Column và chọn tên thuộc tính là Start Time Day.

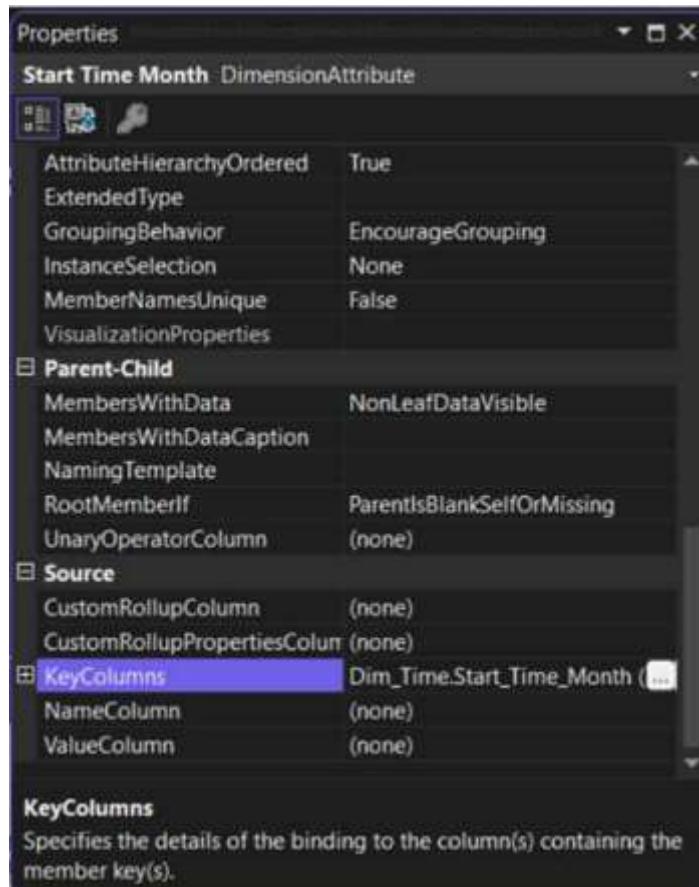
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- **Bước 7:** Chính khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính Start Time Month. Vì thuộc tính Start Time Month là thuộc tính cấp nhỏ hơn Start Time Quarter, Start Time Year nên sẽ lấy khóa cột gồm chính nó và những thuộc tính cấp cao hơn.

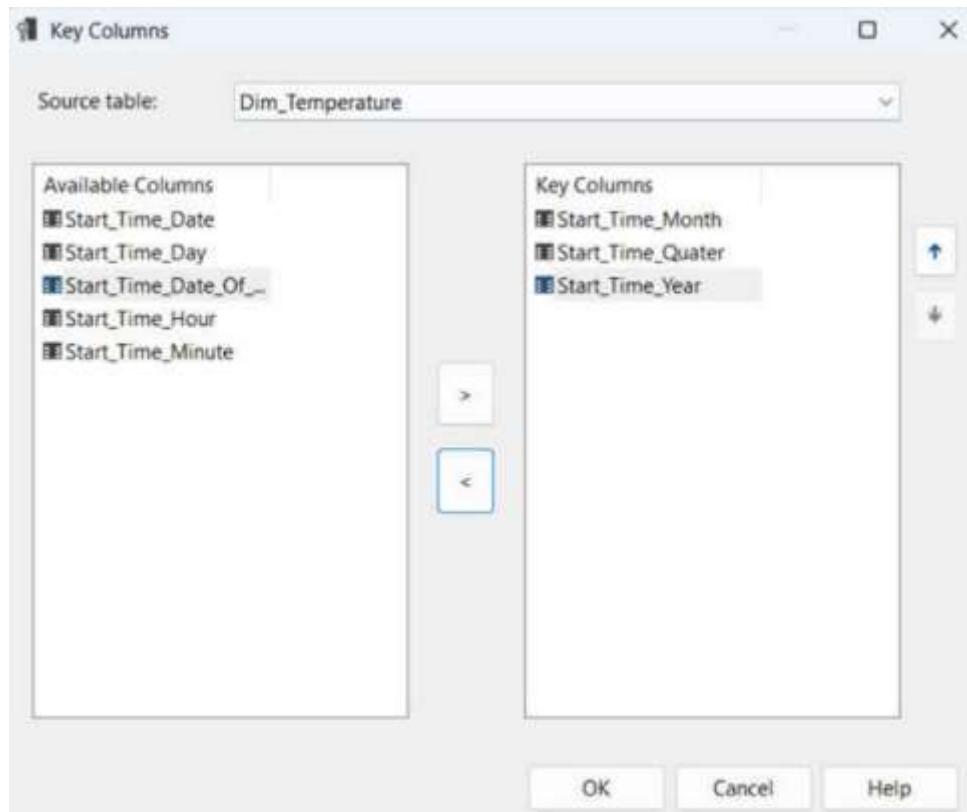
- Tại cửa sổ Properties của thuộc tính Start Time Date, chọn KeyColumns.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



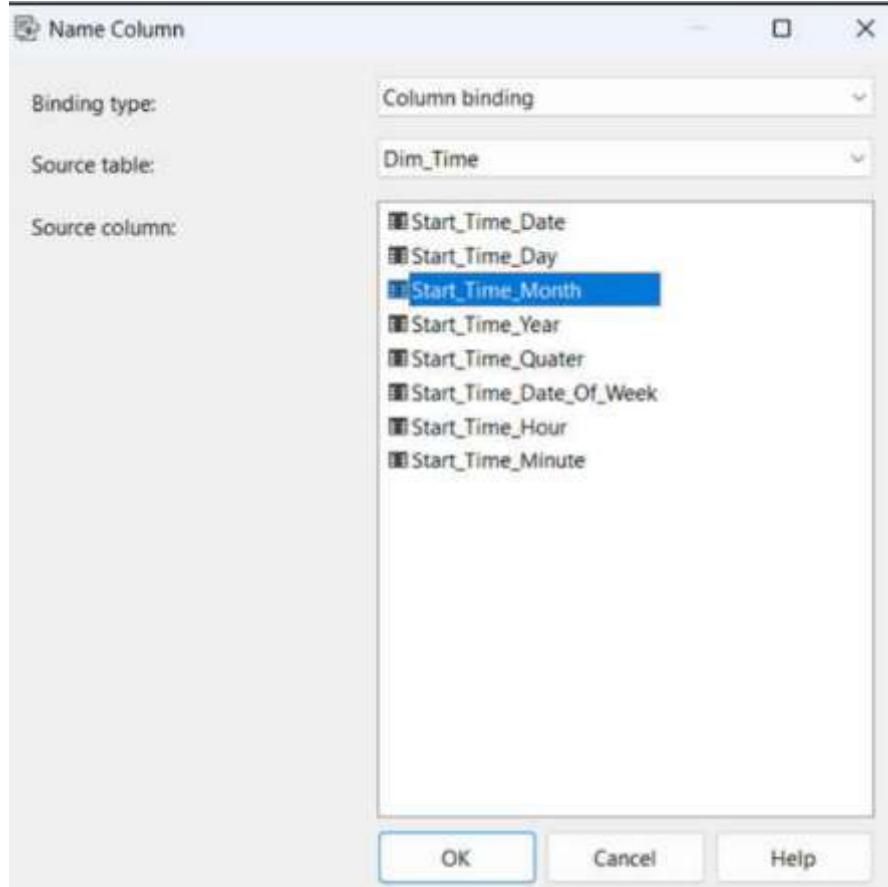
- Thêm các những thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để

hoàn tất.



## Đồ án xây dựng kho dữ liệu US ACCIDENTS

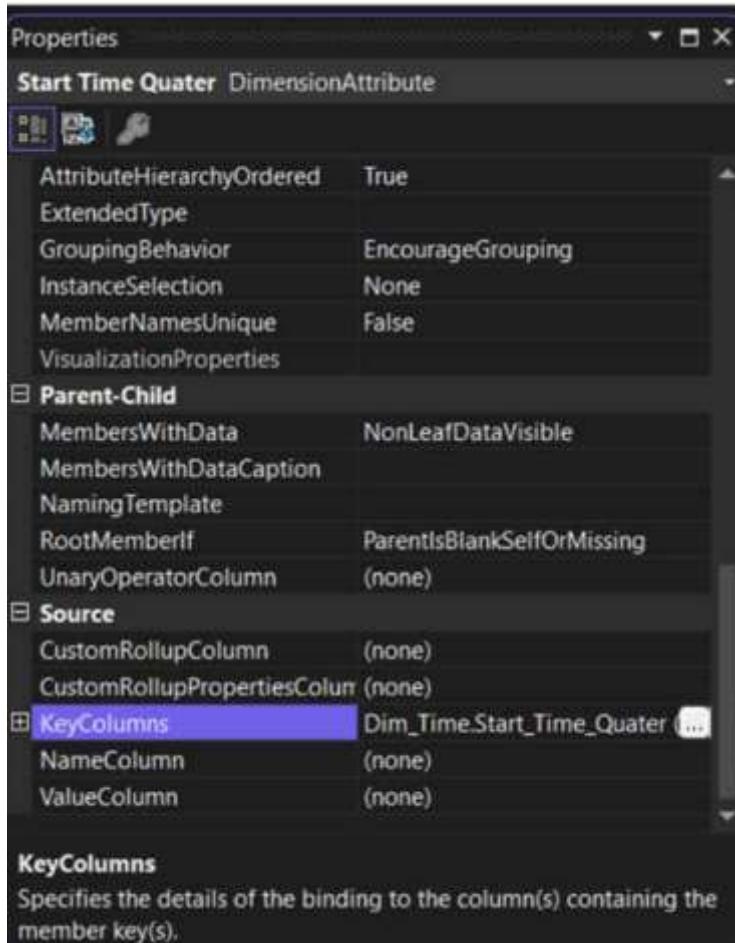
- Tại cửa sổ Properties của thuộc tính Start Time Month, ta chọn Name Column và chọn tên thuộc tính là Start Time Month.



- **Bước 8:** Chính khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính Start Time Quarter. Vì thuộc tính Start Time Quarter là thuộc tính cấp nhỏ hơn Start Time Year nên sẽ lấy khóa cột gồm chính nó và thuộc tính cấp cao hơn.

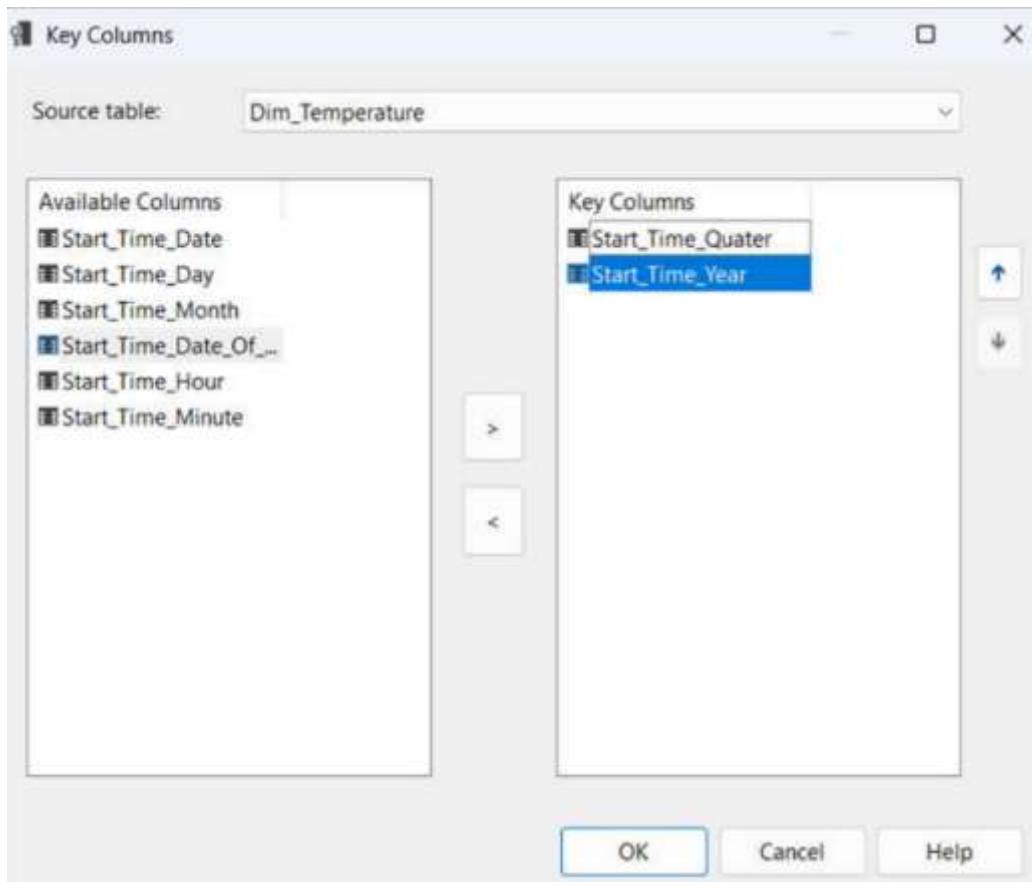
- Tại cửa sổ Properties của thuộc tính Start Time Quarter, chọn KeyColumns.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



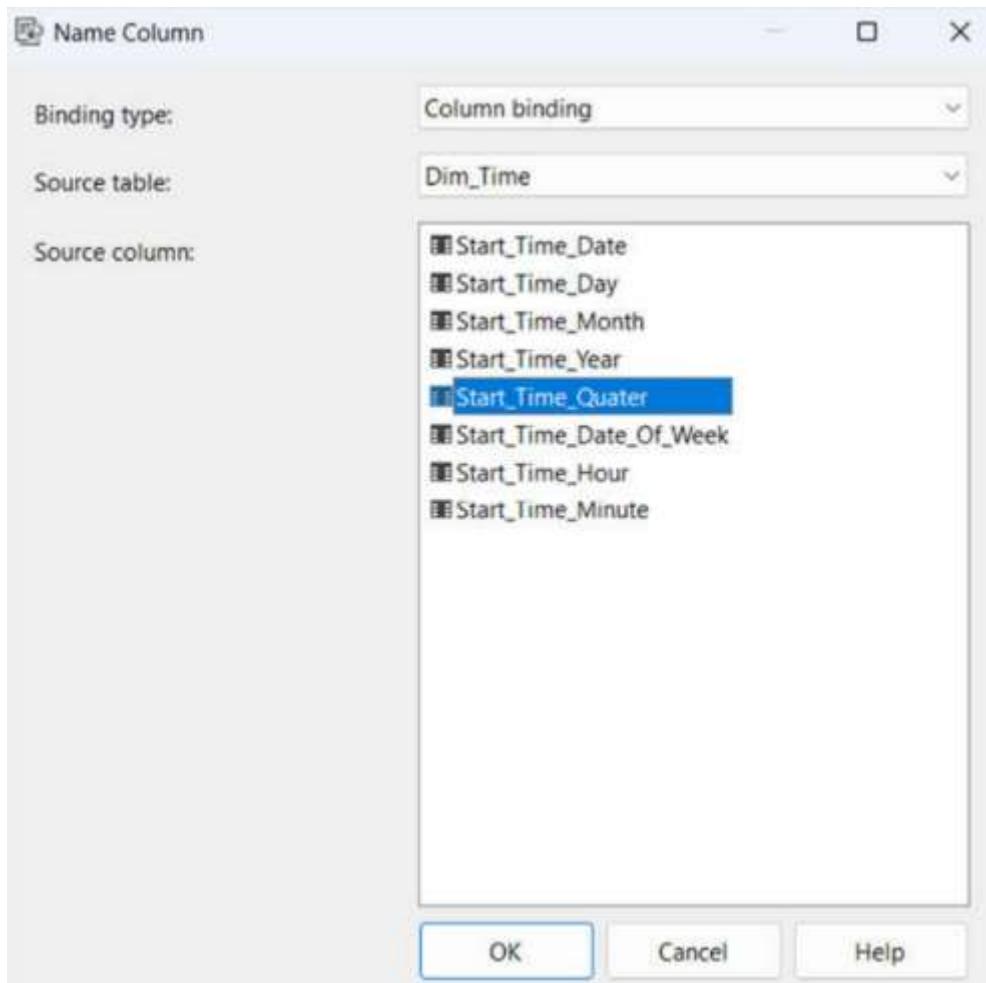
- Thêm các những thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để hoàn tất.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



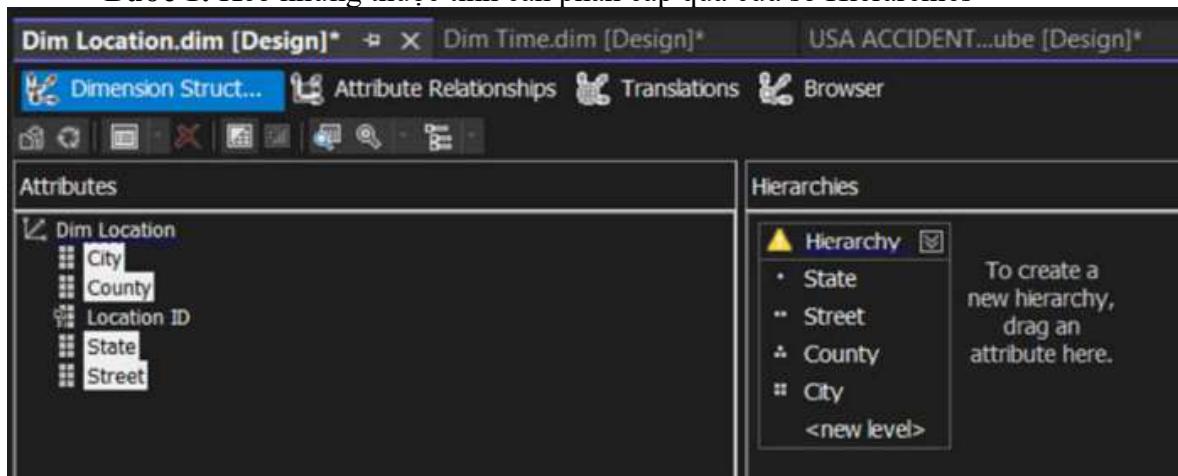
- Tại cửa sổ Properties của thuộc tính Start Time Quarter, ta chọn Name Column và chọn tên thuộc tính là Start Time Quarter.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



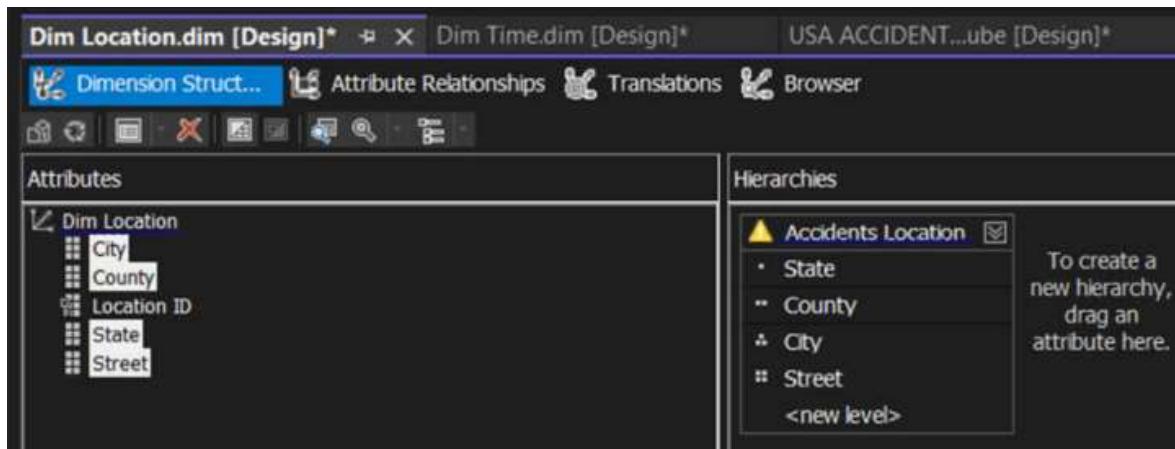
### 5.2. PHÂN CẤP TRONG BẢNG Dim\_Location

- Bước 1: Kéo những thuộc tính cần phân cấp qua cửa sổ Hierarchies

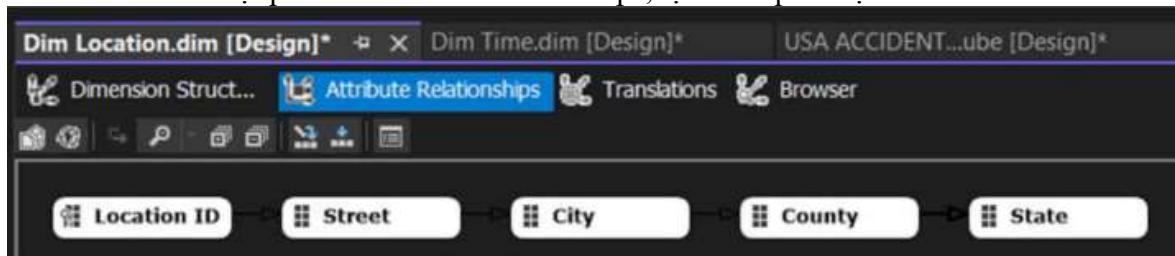


- Bước 2: Sắp xếp lại các thuộc tính phân cấp theo thứ tự: State -> County -> City -> Street và đổi tên Hierarchies thành Accidents Location.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

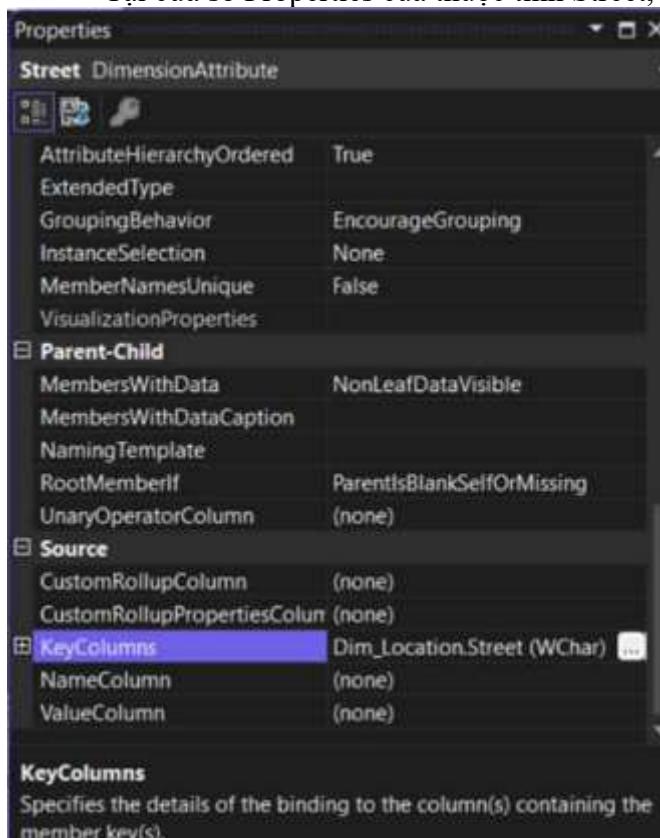


- **Bước 3:** Tại panel Attribute Relationships, tạo mối quan hệ như sau



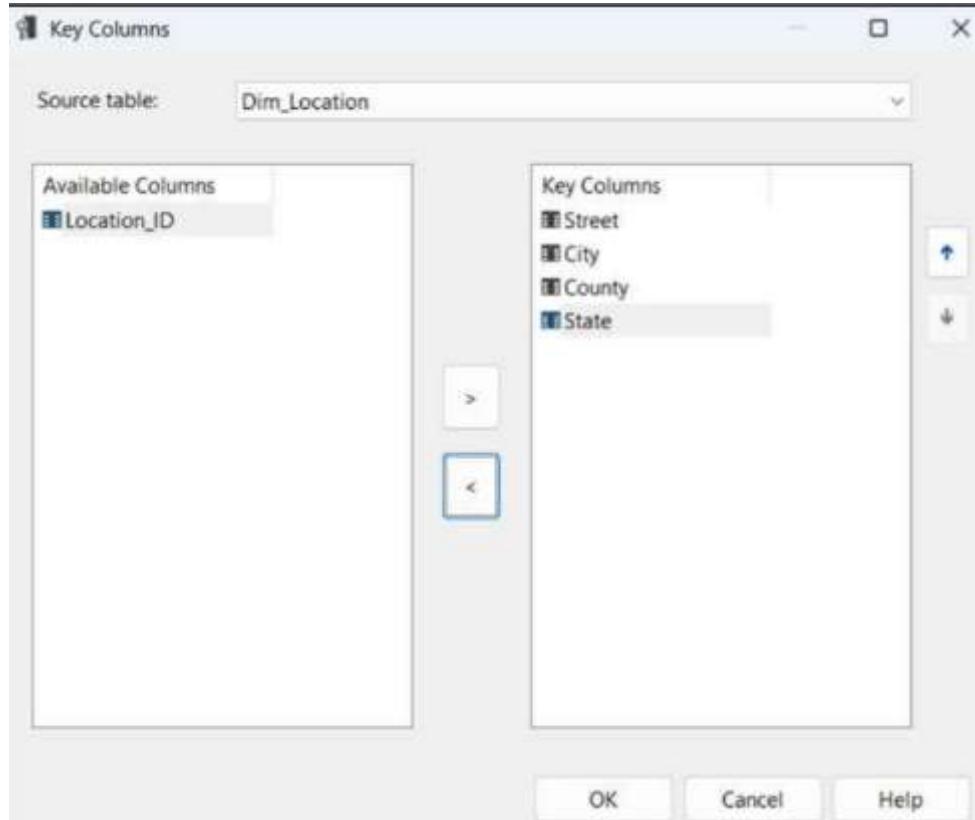
- **Bước 4:** Chỉnh khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính Street. Vì thuộc tính Street là thuộc tính cấp nhỏ nhất sẽ lấy khóa cột gồm chính nó và những thuộc tính cấp cao hơn.

- Tại cửa sổ Properties của thuộc tính Street, chọn KeyColumns.



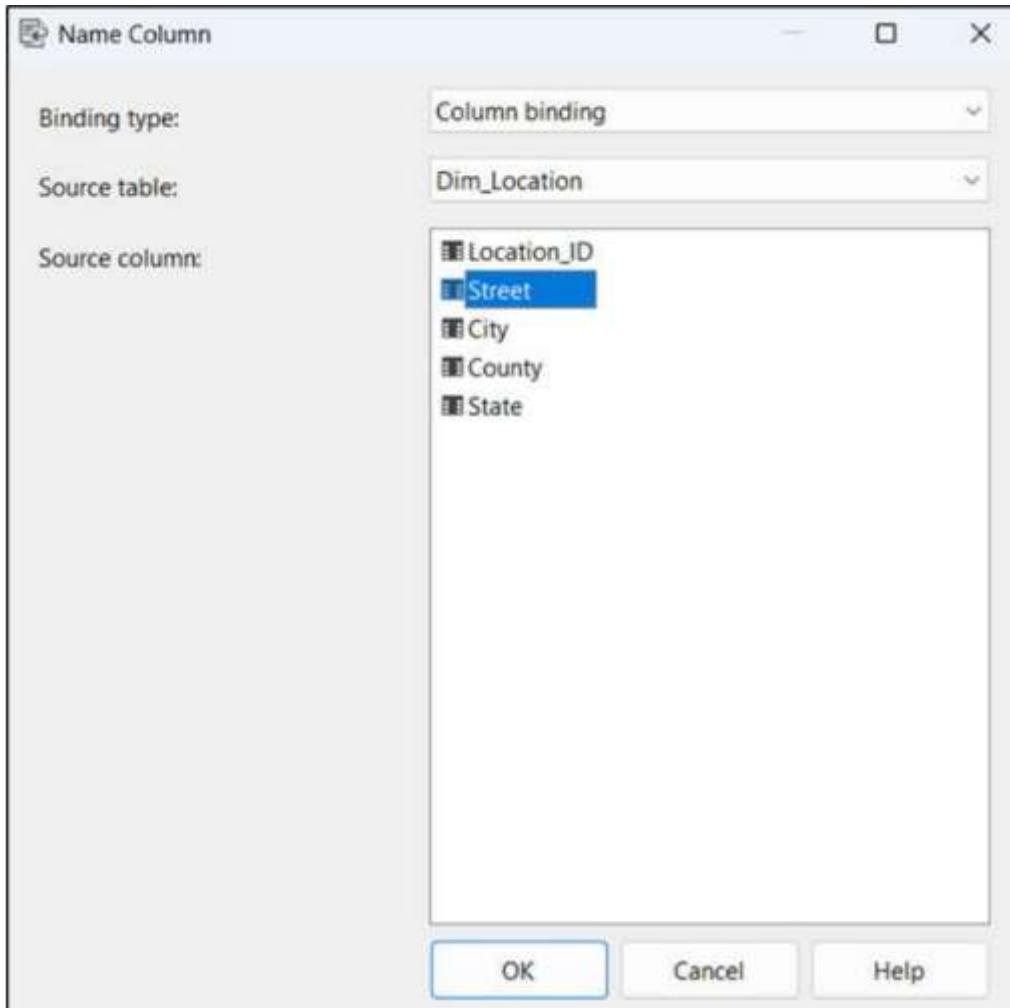
## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

- Thêm các những thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để hoàn tất.



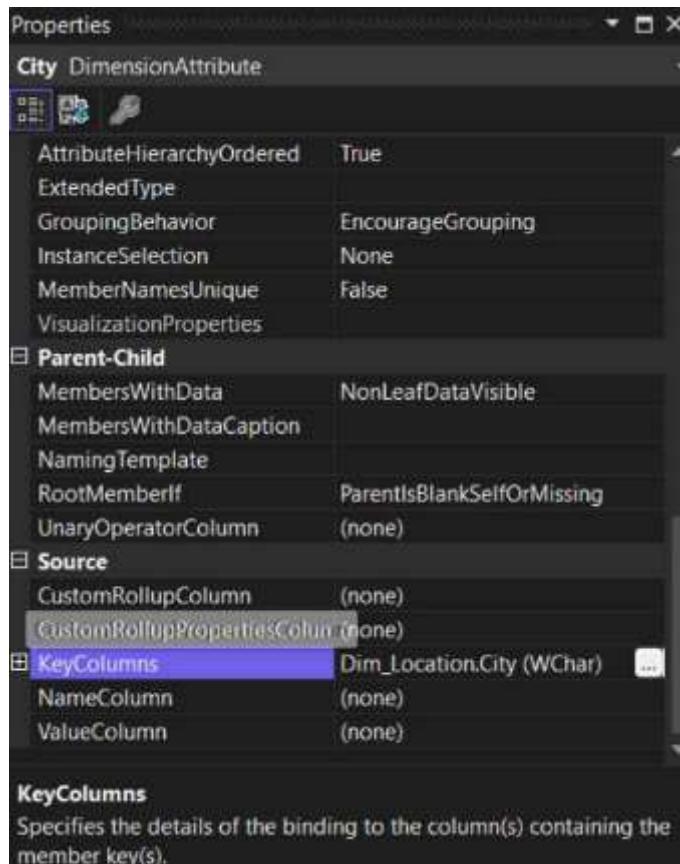
- Tại cửa sổ Properties của thuộc tính Street, ta chọn Name Column và chọn tên thuộc tính là Street.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

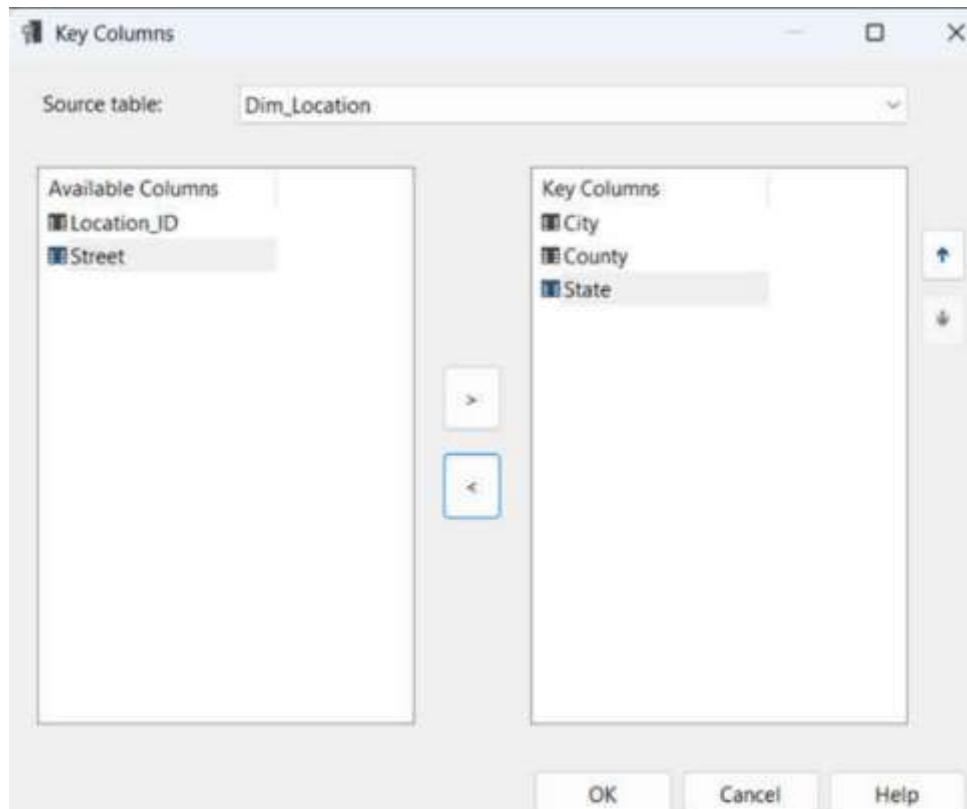


- **Bước 5:** Chỉnh khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính City. Vì thuộc tính City là thuộc tính cấp nhỏ hơn County, State nên sẽ lấy khóa cột gồm chính nó và những thuộc tính cấp cao hơn.
  - Tại cửa sổ Properties của thuộc tính City, chọn KeyColumns.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

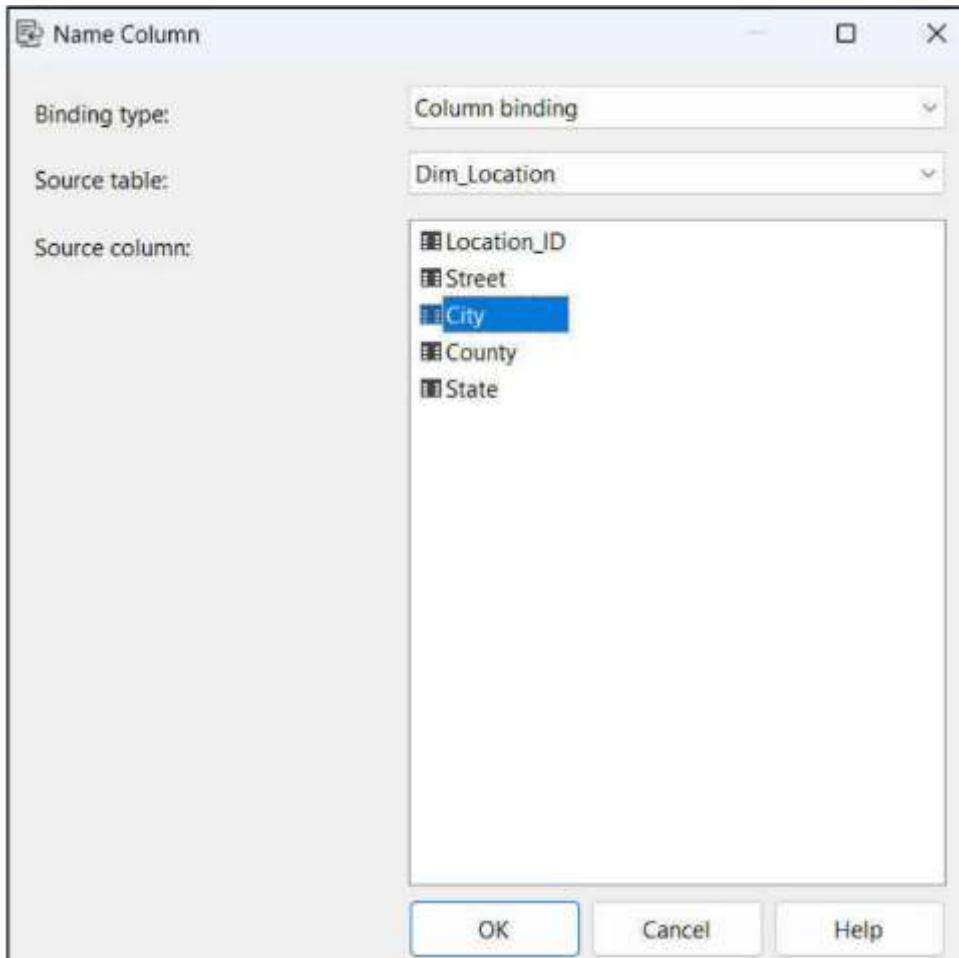


- Thêm các những thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để hoàn tất.



- Tại cửa sổ Properties của thuộc tính City, ta chọn Name Column và chọn tên thuộc tính là City.

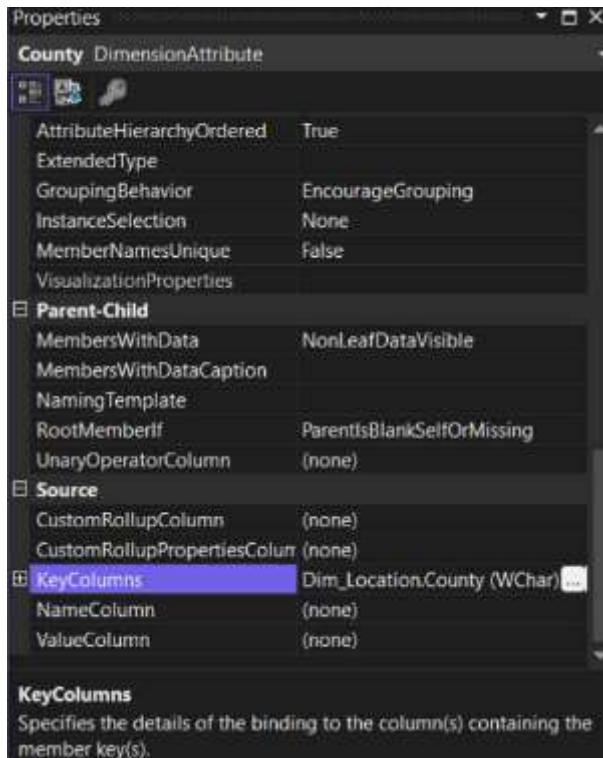
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



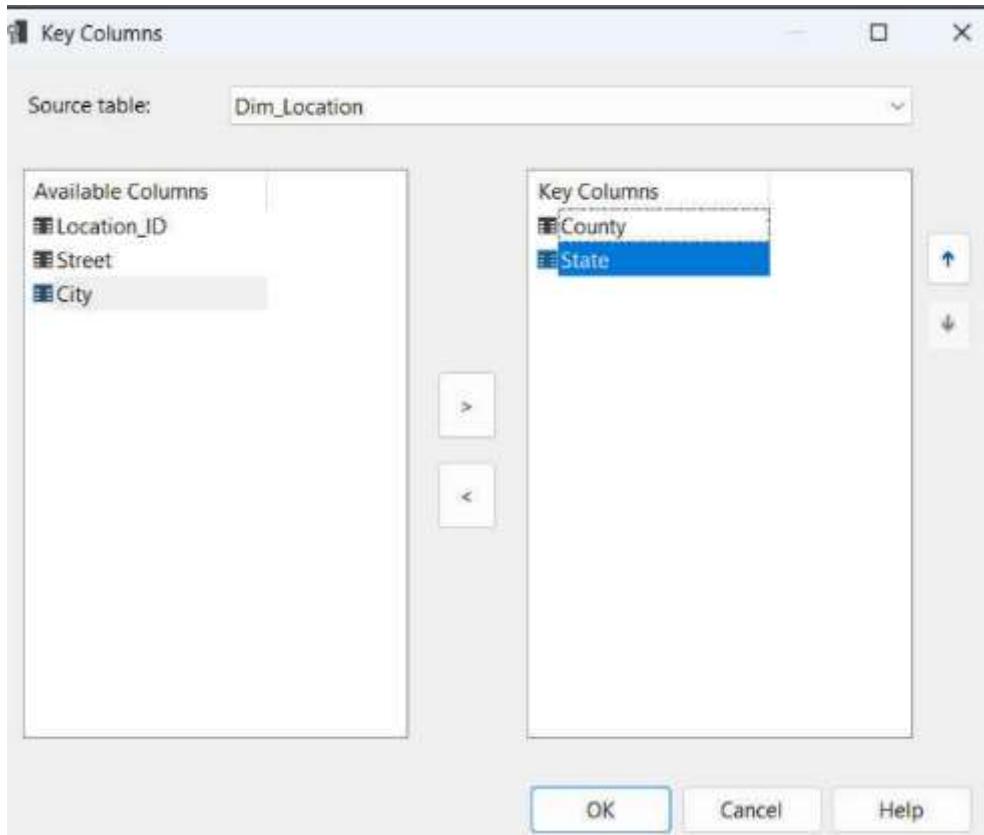
- **Bước 6:** Chỉnh khóa cột (KeyColumns) và tên cột (Name Column) của thuộc tính County. Vì thuộc tính County là thuộc tính cấp nhỏ hơn State nên sẽ lấy khóa cột gồm chính nó và thuộc tính cấp cao hơn.

- Tại cửa sổ Properties của thuộc tính County, chọn KeyColumns.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

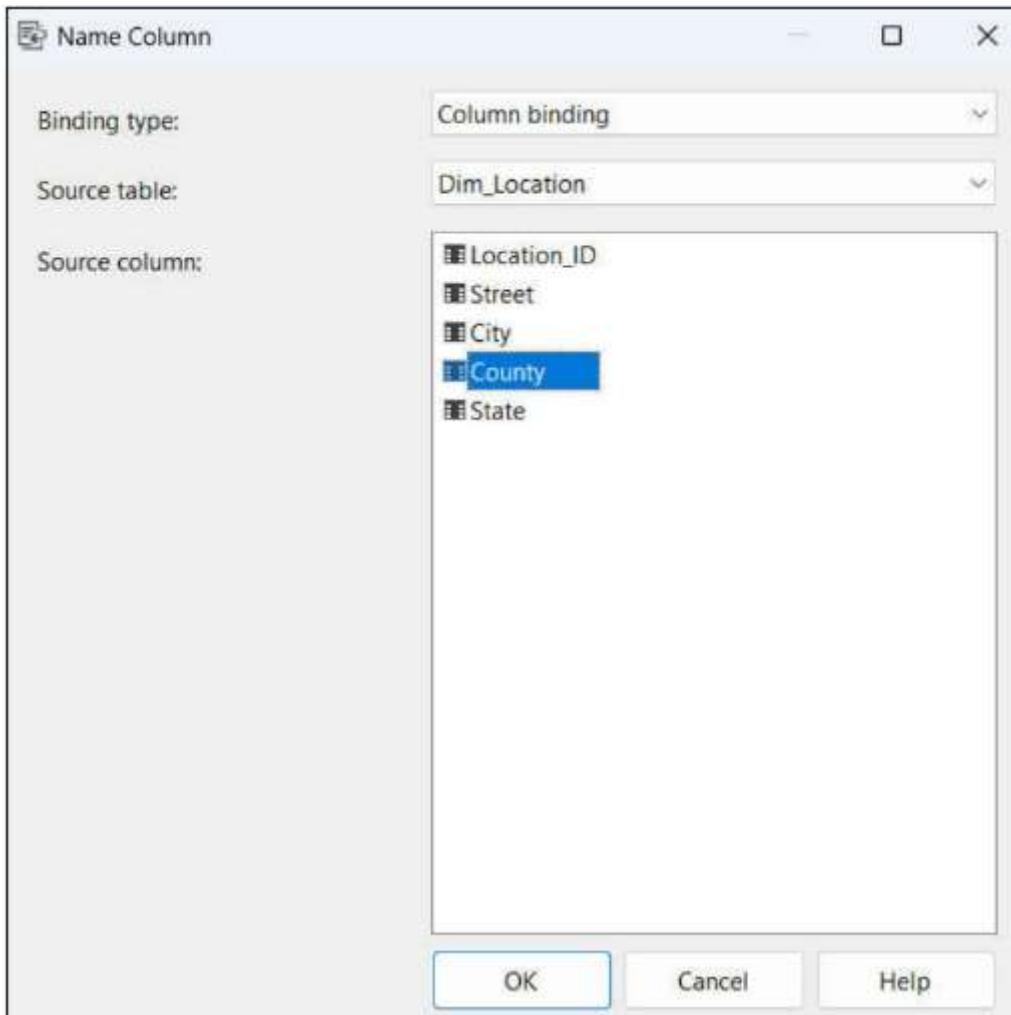


- Thêm các những thuộc tính cấp cao hơn vào KeyColumns, sau đó chọn OK để hoàn tất.



- Tại cửa sổ Properties của thuộc tính County, ta chọn Name Column và chọn tên thuộc tính là County.

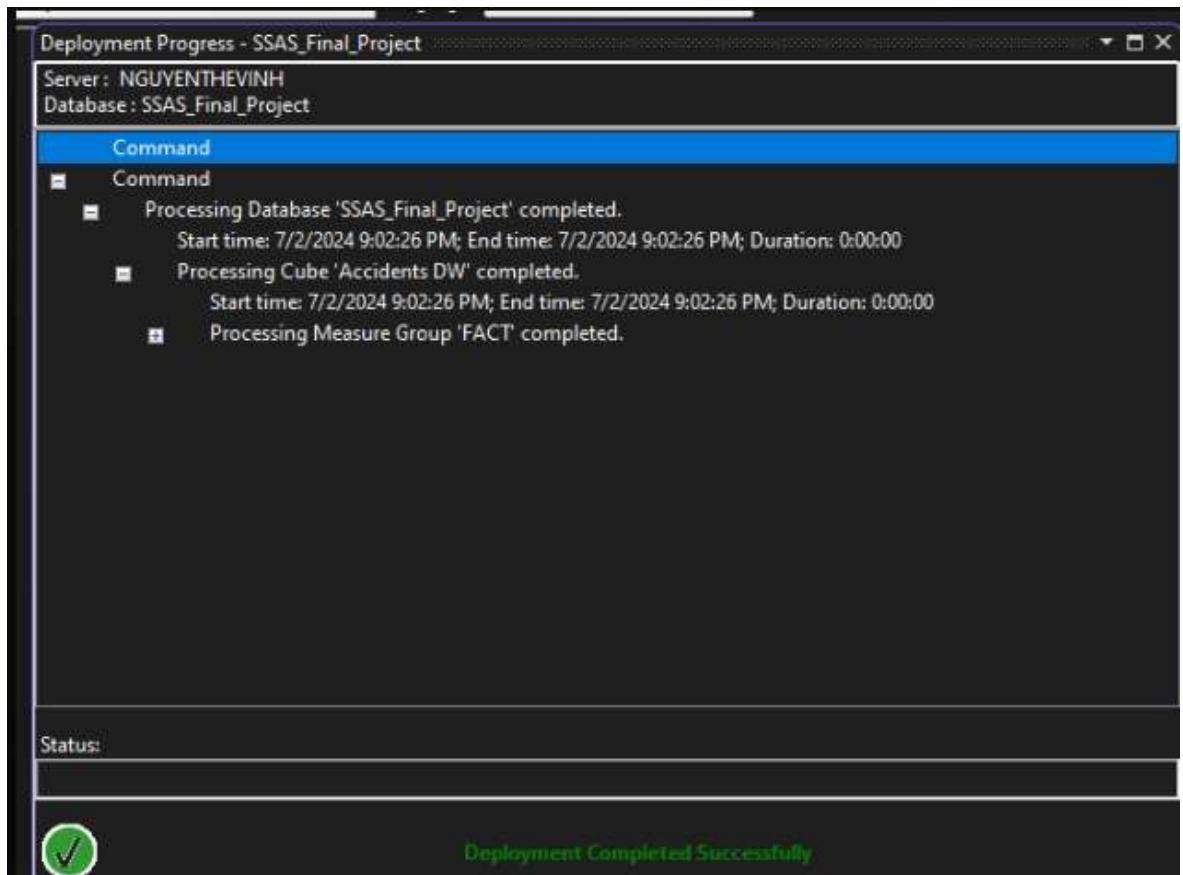
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



### 5.3. CHẠY DỰ ÁN SSAS

- Sau khi quá trình phân cấp cho các bảng chiều hoàn tất, ta thực hiện delpoy project để đảm bảo không có lỗi xảy ra sau quá trình phân cấp. Nhấn chuột phải vào tên project (Accidents\_DW) và nhấn Deploy.
- Khi deploy thành công, hệ thống sẽ hiển thị như hình sau và chúng ta bắt đầu thực hiện các câu truy vấn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



## 6. THỰC HIỆN 10 CÂU TRUY VẤN (MDX)

### 6.1. Hiển thị danh sách các tiểu bang có dữ liệu tai nạn theo từng năm.

#### 6.1.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

The screenshot shows the SSAS Data Source View (DSV) interface. On the left, the cube structure is displayed under the 'Accidents DW' cube. It includes measures like 'FACT Count' and dimensions such as 'Dim Location' which further breaks down into City, County, Location ID, State, Street, and Hierarchy\_Location. On the right, a fact table is shown with the following data:

Start Time Year	State	FACT Count
2022	AL	1224
2022	AR	1329
2022	AZ	4744
2022	CA	46838
2022	CO	2756
2022	CT	2208
2022	DC	1034
2022	DE	232
2022	FL	24473
2022	GA	4597
2022	IA	705
2022	ID	236
2022	IL	2888
2022	IN	1752
2022	KS	858
2022	KY	136
2022	LA	3582
2022	MA	123
2022	MD	5260
2022	MI	3474
2022	MN	7417
2022	MO	2153
2022	MS	276
2022	MT	3514
2022	NC	7692
2022	ND	247
2022	NE	141
2022	NJ	3418
2022	NM	117
2022	NV	538
2022	NY	7840

### 6.1.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT
    NON EMPTY [Measures].[FACT Count] on COLUMNS,
    NON EMPTY [Dim Date].[Start Time Year].[Start Time Year].MEMBERS * [Dim Location].[STATE].MEMBERS on ROWS
FROM [Accidents Dw]
```

- Kết quả:

## *Đồ án xây dựng kho dữ liệu US ACCIDENTS*

		Messages	Results
		FACT Count	
2022	All	216108	
2022	AL	1224	
2022	AR	1329	
2022	AZ	4744	
2022	CA	46838	
2022	CO	2756	
2022	CT	2208	
2022	DC	1034	
2022	DE	232	
2022	FL	24473	
2022	GA	4597	
2022	IA	705	
2022	ID	236	
2022	IL	2888	
2022	IN	1752	
2022	KS	858	
2022	KY	136	
2022	LA	3582	
2022	MA	123	
2022	MD	5260	
2022	MI	3474	
2022	MN	7417	
2022	MO	2153	
2022	MS	276	
2022	MT	3514	
2022	NC	7692	
2022	ND	247	
2022	NE	141	
2022	NJ	3418	
2022	NM	117	

### **6.1.3. Thực hiện trên Excel.**

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotChart Fields ▾ ×

Choose fields to add to report:

Search 

**Σ FACT**

- Average Severity
- FACT Count**
- Sum Distance

**Dim Airport**

- Airport Code

Drag fields between areas below:

 FILTERS	 LEGEND (SE... Start Time... ▾)
 AXIS (CATE... State ▾)	 VALUES FACT Cou... ▾)

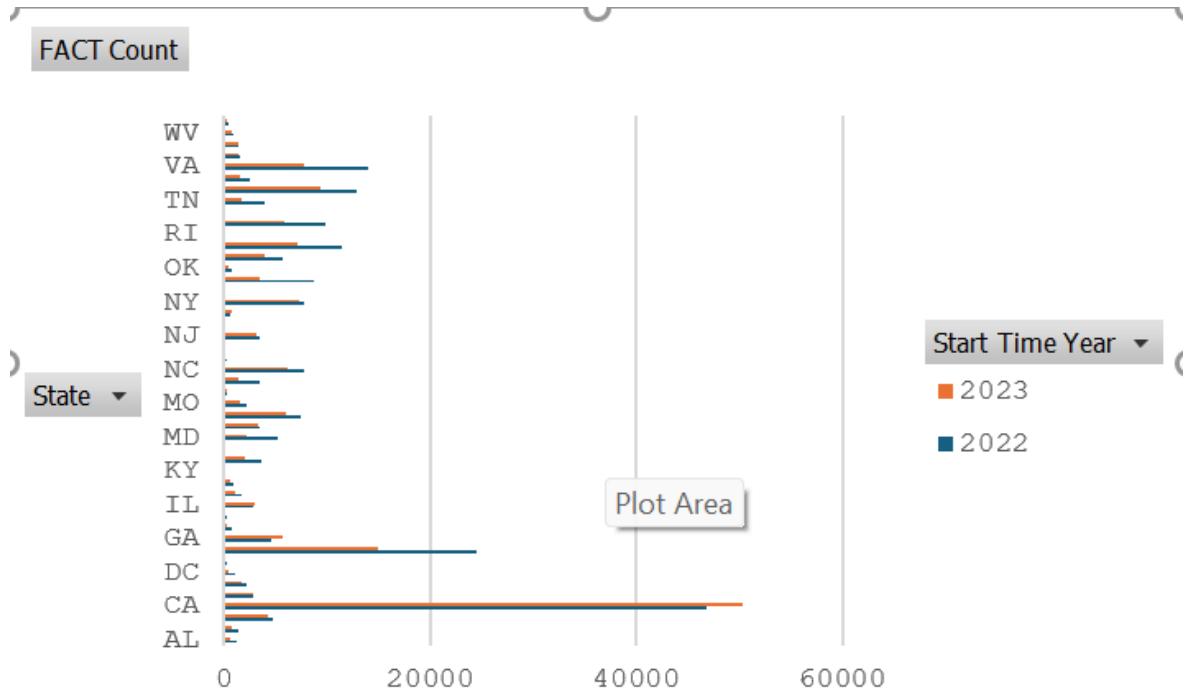
- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

*Đồ án xây dựng kho dữ liệu US ACCIDENTS*

FACT Count	Column Labels	2022	2023	Grand Total
Row Labels				
AL		1224	611	1835
AR		1329	765	2094
AZ		4744	4180	8924
CA		46838	50287	97125
CO		2756	2882	5638
CT		2208	1712	3920
DC		1034	471	1505
DE		232	106	338
FL		24473	14959	39432
GA		4597	5774	10371
IA		705	322	1027
ID		236	178	414
IL		2888	2994	5882
IN		1752	1015	2767
KS		858	643	1501
KY		136	52	188
LA		3582	2056	5638
MA		123	77	200
MD		5260	2152	7412
MI		3474	3356	6830
MN		7417	6075	13492
MO		2153	1560	3713

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



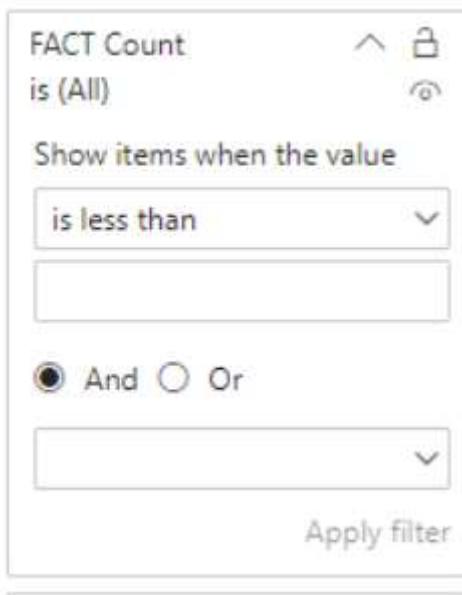
### 6.1.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

The screenshot shows the PowerBI interface with the "Filters" pane open. The "Filters" pane displays three sections: "Filters on this visual", "Filters on this page", and "Filters on all pages". Under "Filters on this visual", three filters are applied: "FACT Count is (All)", "Start Time Year is (All)", and "State is (All)". Under "Filters on this page", there is an "Add data fields here" button. Under "Filters on all pages", there is also an "Add data fields here" button. To the right of the filters, there is a "Visualizations" pane showing various visualization icons, and a "Columns" pane listing "State", "Start Time Year", and "FACT Count". At the bottom of the filters pane, there are "Drill through", "Cross-report", and "Keep all filters" settings.

## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.



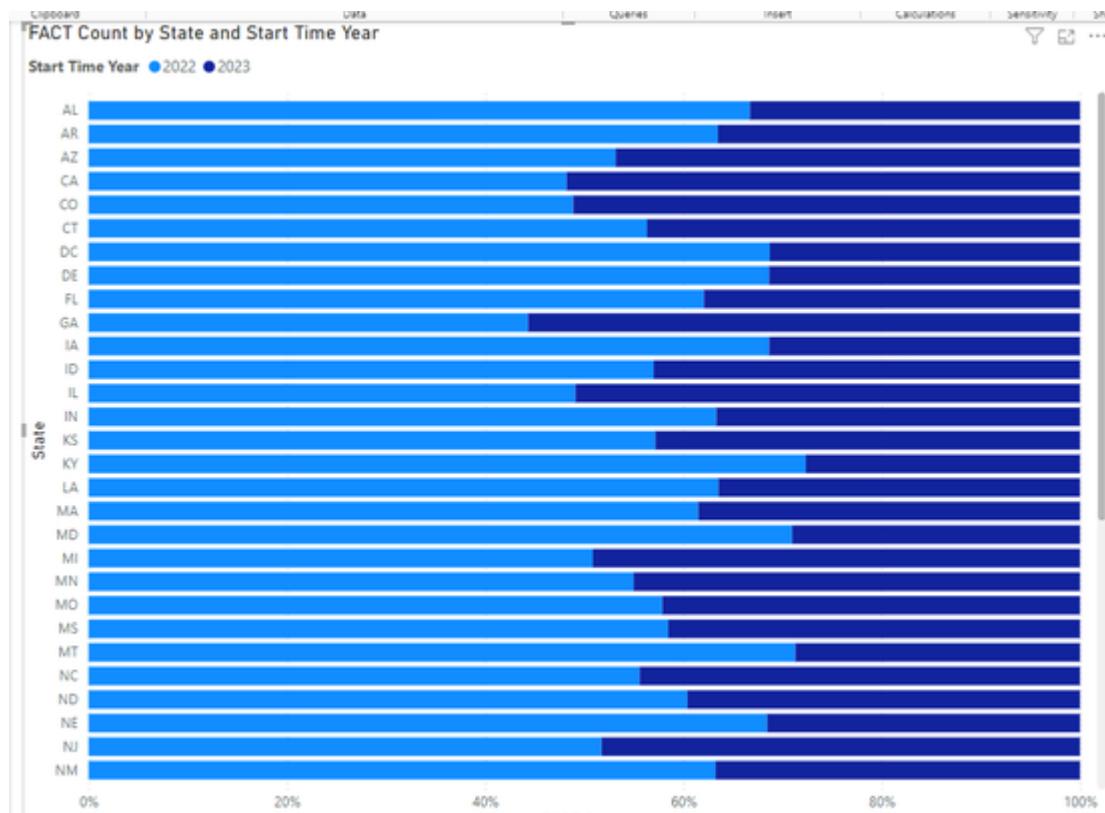
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

State	Start Time	Year	FACT_Co
AL		2022	1224
AL		2023	611
AR		2022	1329
AR		2023	765
AZ		2022	4744
AZ		2023	4180
CA		2022	46838
CA		2023	50287
CO		2022	2756
CO		2023	2882
CT		2022	2208
CT		2023	1712
DC		2022	1034
DC		2023	471
DE		2022	232
DE		2023	106
FL		2022	24473
FL		2023	14959
GA		2022	4597
GA		2023	5774
IA		2022	705
IA		2023	322
ID		2022	236
ID		2023	178
IL		2022	2888
IL		2023	2994
IN		2022	1752
IN		2023	1015
KS		2022	858
KS		2023	643
KY		2022	136
KY		2023	52
LA		2022	3582
Total			382660

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.

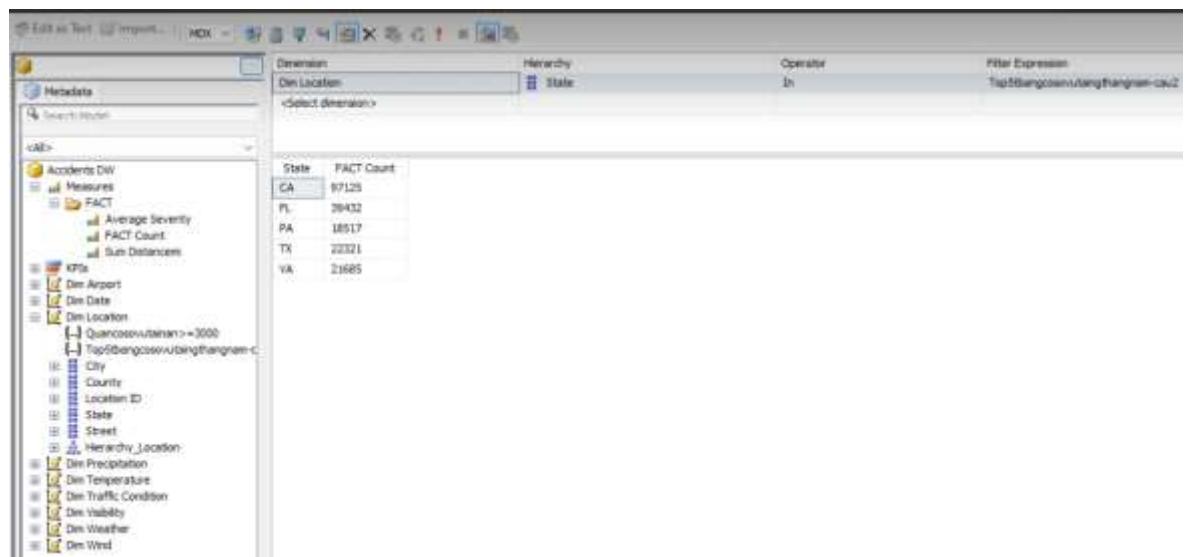
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



## 6.2. Top 5 tiểu bang có số vụ tai nạn lớn nhất của từng năm, sắp xếp giảm dần (TOP COUNT).

### 6.2.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.



## *Đồ án xây dựng kho dữ liệu US ACCIDENTS*

### 6.2.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT  
NON EMPTY {[Measures].[FACT Count]} ON COLUMNS,  
{TOPCOUNT([Dim Location].[State].[State],5,[Measures].[FACT Count])} ON ROWS  
FROM [Accidents DW]
```

- Kết quả:

	FACT Count
CA	97125
FL	39432
TX	22321
VA	21685
PA	18517

### 6.2.3. Thực hiện trên Excel.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotTable Fields ▾ ×

Choose fields to add to report:

Search

▲ **Σ FACT**

- Average Severity
- FACT Count
- Sum Distance

▲ **Dim Airport**

- Airport Code

Drag fields between areas below:

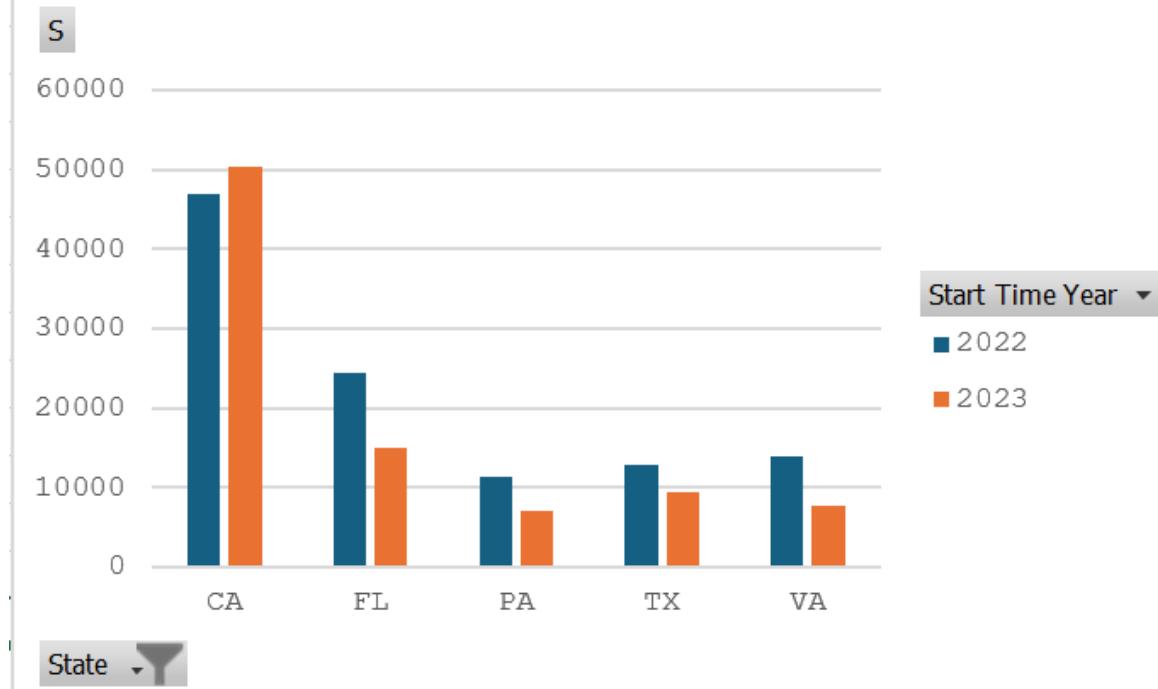
FILTERS	COLUMNS
	Start Time... ▾
ROWS	VALUES
State ▾	FACT Cou... ▾

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

A	B	C	D
FACT Count	Column Labels		
Row Labels	2022	2023	Grand Total
CA	46838	50287	97125
FL	24473	14959	39432
PA	11397	7120	18517
TX	12906	9415	22321
VA	13967	7718	21685
<b>Grand Total</b>	<b>109581</b>	<b>89499</b>	<b>199080</b>

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.



### 6.2.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

The screenshot shows the 'Filters' pane in Power BI. It includes sections for 'Filters on this visual', 'Filters on this page', and 'Filters on all pages'. Under 'Filters on this visual', there are three items: 'State' (top 5 by Sum of Location), 'FACT Count' (is (All)), and 'Start Time Year' (is (All)). Below these is a button 'Add data fields here'. The 'Visualizations' pane on the right displays a grid of icons representing different chart types like bar charts, line graphs, and maps.

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.

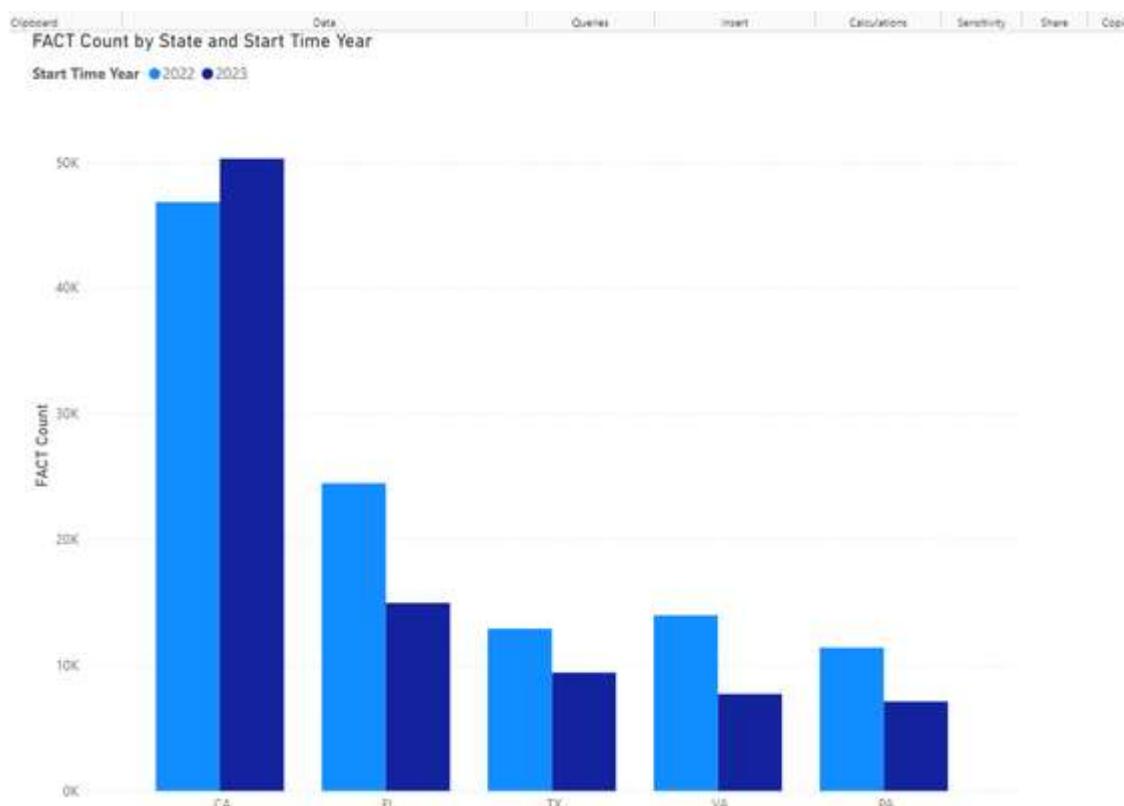
The screenshot shows the filter settings for 'FACT Count'. The current value is 'is (All)'. The dropdown 'Show items when the value' is set to 'is less than'. The radio button 'And' is selected. At the bottom is a 'Apply filter' button.

- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

State	Start Time Year	FACT Count
CA	2023	50287
CA	2022	46838
FL	2022	24473
FL	2023	14959
VA	2022	13967
TX	2022	12906
PA	2022	11397
TX	2023	9415
VA	2023	7718
PA	2023	7120
<b>Total</b>		<b>199080</b>

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.



### 6.3. Liệt kê các loại thời tiết có số vụ tai nạn từ 1000 trở xuống.

#### 6.3.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Dimension	Hierarchy	Operator	Filter Expression
Dim Weather	Weather Condition	In	Loai tho tiet co vua han < 1000-cau3
<Select dimension>			
Weather Condition	FACT Count		
Blowing Dust	3		
Blowing Dust / Windy	20		
Blowing Snow	199		
Blowing Snow / Windy	489		
Drifting Snow / Windy	4		
Drizzle	298		
Drizzle / Windy	1		
Drizzle and Fog	57		
Fog / Windy	106		
Freezing Drizzle	5		
Freezing Rain	18		
Freezing Rain / Windy	4		
Funnel Cloud	1		
Hail	4		
Haze / Windy	400		
Heavy Drizzle	18		
Heavy Rain / Windy	172		
Heavy Sleet	4		
Heavy Sleet / Windy	1		
Heavy Sleet and Thunder	1		
Heavy Snow	872		
Heavy Snow / Windy	264		
Heavy Snow with Thunder	1		
Heavy T-Storm	256		
Heavy T-Storm / Windy	22		
Light Drizzle / Windy	22		
Light Freezing Drizzle	56		
Light Freezing Rain	280		
Light Freezing Rain / Windy	11		
Light Rain / Windy	865		
Light Rain Shower	11		

### 6.3.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT {[Measures].[Fact Count]} ON COLUMNS,
       [Dim Weather].[Weather Condition]. MEMBERS HAVING [Measures].[FACT Count] <= 1000 ON ROWS
  FROM [Accidents DW]
```

- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

	FACT Count
Blowing Dust	3
Blowing Dust / Windy	20
Blowing Snow	199
Blowing Snow / Windy	489
Drifting Snow / Windy	4
Drizzle	298
Drizzle / Windy	1
Drizzle and Fog	57
Fog / Windy	106
Freezing Drizzle	5
Freezing Rain	18
Freezing Rain / Windy	4
Funnel Cloud	1
Hail	4
Haze / Windy	400
Heavy Drizzle	18
Heavy Rain / Windy	172
Heavy Sleet	4
Heavy Sleet / Windy	1
Heavy Sleet and Thunder	1
Heavy Snow	872
Heavy Snow / Windy	264
Heavy Snow with Thunder	1
Heavy T-Storm	256
Heavy T-Storm / Windy	22
Light Drizzle / Windy	22
Light Freezing Drizzle	56
Light Freezing Rain	280
Light Freezing Rain / Windy	11
Light Rain / Windy	865

### 6.3.3. Thực hiện trên Excel.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotTable Fields

Choose fields to add to report:

Search 

FACT Count

Average Severity

Sum Distance

Dim Airport

Airport Code

Airnort ID

Drag fields between areas below:

FILTERS	COLUMNS

ROWS	VALUES
Weather Condition ▾	FACT Count ▾

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

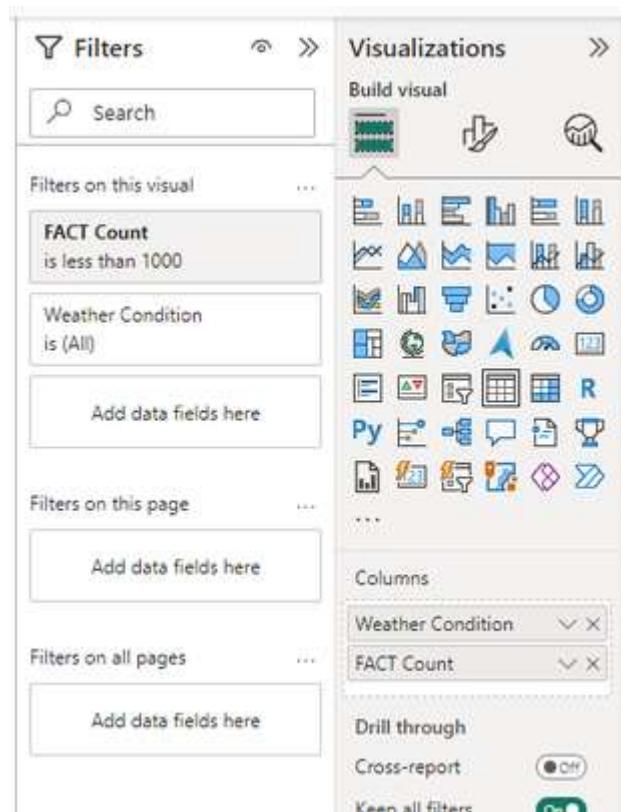
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Row Labels	FACT Count
Blowing Dust	3
Blowing Dust / Windy	20
Blowing Snow	199
Blowing Snow / Windy	489
Drifting Snow / Windy	4
Drizzle	298
Drizzle / Windy	1
Drizzle and Fog	57
Fog / Windy	106
Freezing Drizzle	5
Freezing Rain	18
Freezing Rain / Windy	4
Funnel Cloud	1
Hail	4
Haze / Windy	400
Heavy Drizzle	18
Heavy Rain / Windy	172
Heavy Sleet	4
Heavy Sleet / Windy	1
Heavy Sleet and Thunder	1
Heavy Snow	872
Heavy Snow / Windy	264
Heavy Snow with Thunder	1
Heavv T-Storm	256

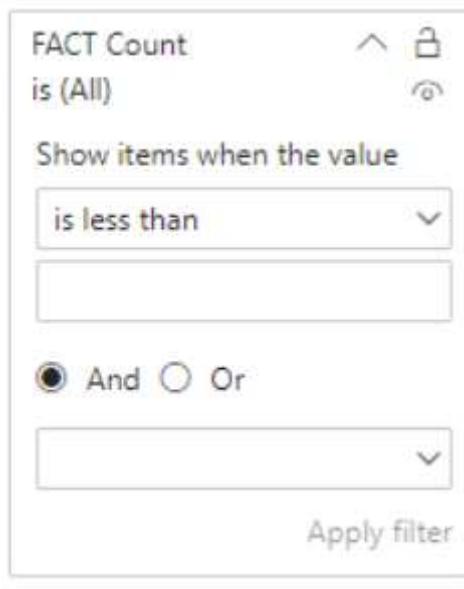
### 6.3.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.



- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Weather Condition	FACT Count
Blowing Dust	3
Blowing Dust / Windy	20
Blowing Snow	199
Blowing Snow / Windy	489
Drifting Snow / Windy	4
Drizzle	298
Drizzle / Windy	1
Drizzle and Fog	57
Fog / Windy	106
Freezing Drizzle	5
Freezing Rain	18
Freezing Rain / Windy	4
Funnel Cloud	1
Hail	4
Haze / Windy	400
Heavy Drizzle	18
Heavy Rain / Windy	172
Heavy Sleet	4
Heavy Sleet / Windy	1
Heavy Sleet and Thunder	1
Heavy Snow	872
Heavy Snow / Windy	264
Heavy Snow with Thunder	1
Heavy T-Storm	256
Heavy T-Storm / Windy	22
Light Drizzle / Windy	22
Light Freezing Drizzle	56
Light Freezing Rain	280
Light Freezing Rain / Windy	11
Light Rain / Windy	865
Light Rain Shower	11
Light Rain Shower / Windy	2
Light Rain with Thunder	248
<b>Total</b>	<b>8268</b>

### 6.4. Hiển thị mức độ nghiêm trọng trung bình từng năm của quận NEW YORK với loại thời tiết là Cloudy.

#### 6.4.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Start Time Year	Average Severity
2022	2.84210526315789
2023	3.12328767123288

### 6.4.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT
    {[Dim Date].[Start Time Year].Members} ON COLUMNS,
    {[Measures].[Average Severity]} ON ROWS
FROM [Accidents DW]
WHERE
    ([Dim Location].[County].[New York], [Dim Weather].[Weather Condition].[Cloudy])
```

- Kết quả:

	All	2022	2023	Unknown
Average Severity	3.02702702702703	2.84210526315789	3.12328767123288	(null)

### 6.4.3. Thực hiện trên Excel.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotTable Fields

Choose fields to add to report:

Search 

FACT

- Average Severity
- FACT Count
- Sum Distance

Dim Airport

- Airport Code
- Airport ID

Drag fields between areas below:

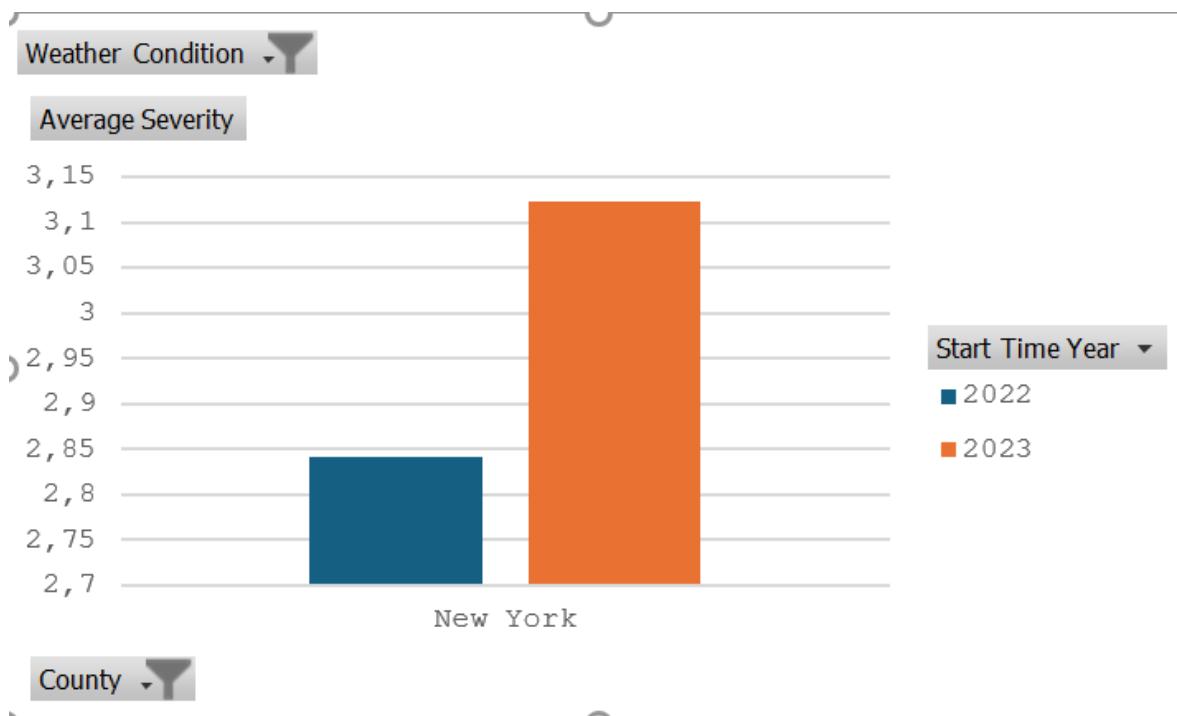
FILTERS	COLUMNS
Weather Condition	Start Time Year
ROWS	SUM VALUES
County	Average Severity

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

A	B	C	D
Weather Condition	Cloudy		
Average Severity	Column Labels		
Row Labels	2022	2023	Grand Total
New York	2,842105263	3,123287671	3,027027027
<b>Grand Total</b>	<b>2,842105263</b>	<b>3,123287671</b>	<b>3,027027027</b>

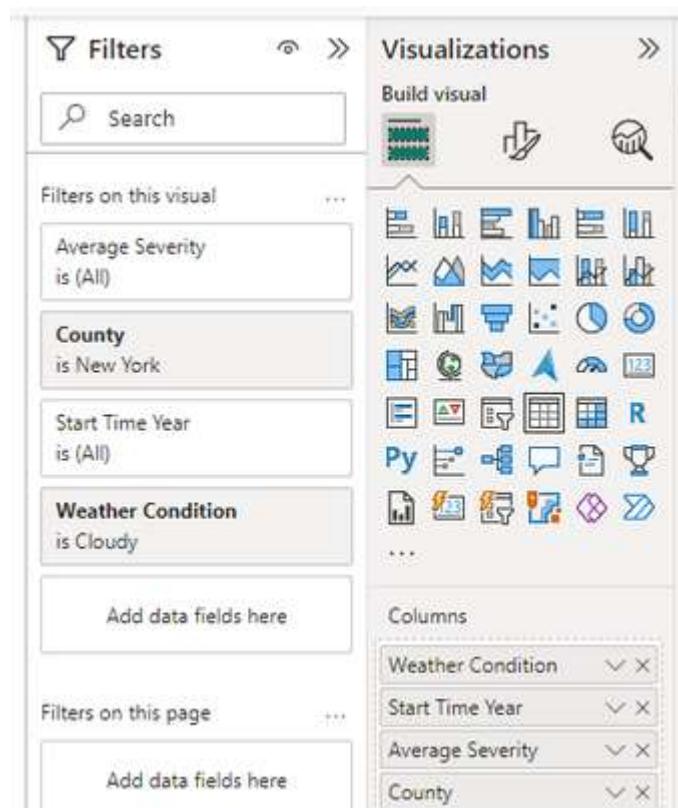
- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.



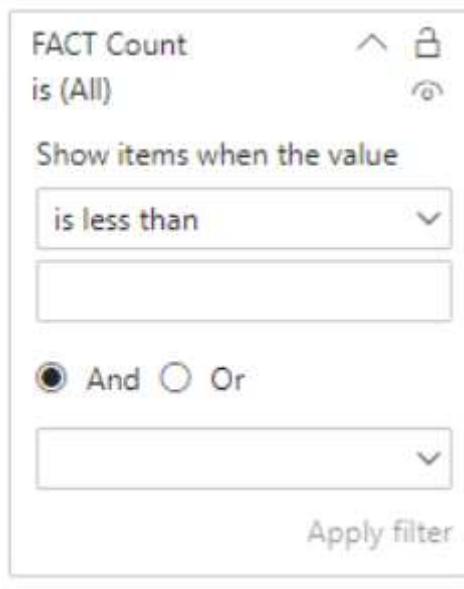
### 6.4.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.

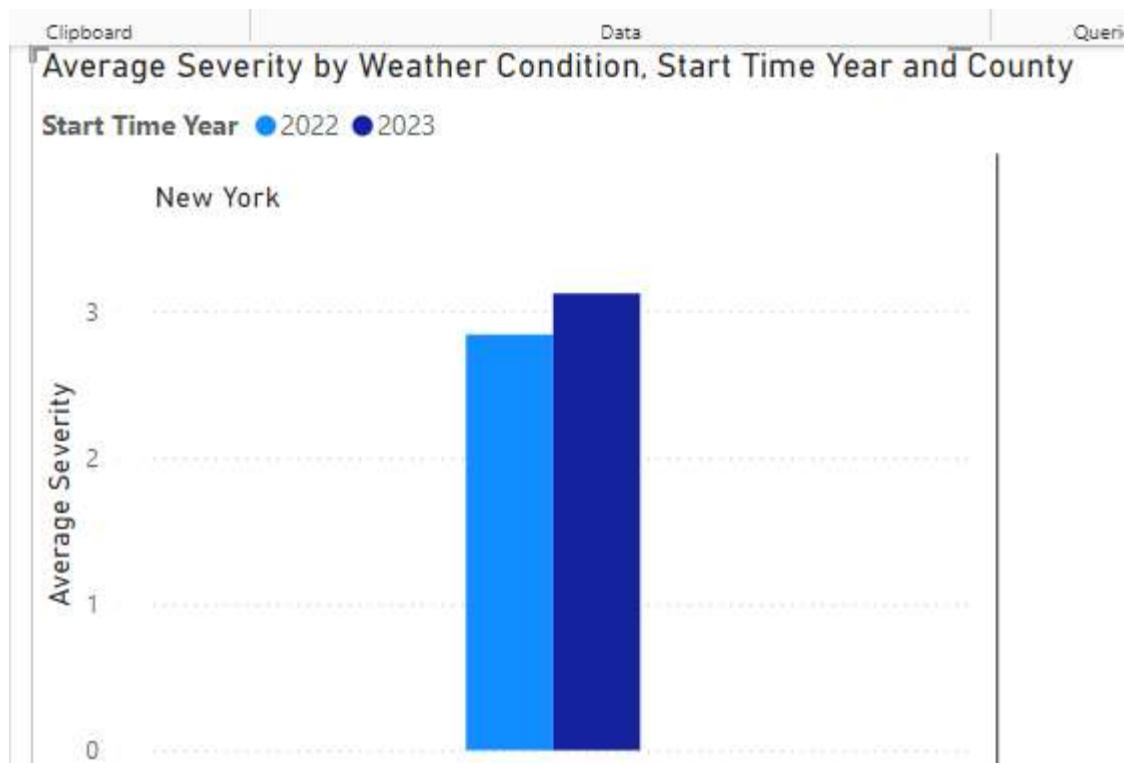


- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Clipboard	Data		
Weather Condition	Start Time Year	Average Severity	County
Cloudy	2022	2.84	New York
Cloudy	2023	3.12	New York
<b>Total</b>		<b>3.03</b>	

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.



### 6.5. Hiển thị mức độ nghiêm trọng trung bình của tai nạn xảy ra từng tháng của từng năm.

#### 6.5.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Dimension	Hierarchy	Operator	Filter Expression
Dim Date	Start Time Month	Equal	
<Select dimension>			
Start Time Year	Start Time Month	Average Severity	
2022	11	3.48057369358329	
2022	12	3.47013997301088	
2023	1	3.57311456534254	
2023	2	3.27939681463911	
2023	3	3.19132420091324	

### 6.5.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT NON EMPTY {[Dim Date].[Start Time Month].Members} ON COLUMNS,
       NON EMPTY GENERATE([Dim Date].[Start Time Year].Members,[Dim Date].[Start Time Year].CurrentMember) ON ROWS
  FROM [Accidents DW]
 WHERE ([Measures].[Average Severity])
```

- Kết quả:

All	1	2	3	11	12
All	3.4649322512319	3.57311456534254	3.27939681463911	3.19132420091324	3.48057369358329
2022	3.47440321104148	(null)	(null)	(null)	3.48057369358329
2023	3.4527397260274	3.57311456534254	3.27939681463911	3.19132420091324	(null)

### 6.5.3. Thực hiện trên Excel.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotTable Fields

Choose fields to add to report:

Search

Average Severity

FACT Count

Sum Distance

Dim Airport

Airport Code

Airnort ID

Drag fields between areas below:

FILTERS	COLUMNS
	Start Time Month
ROWS	Σ VALUES
Start Time Year	Average Severity

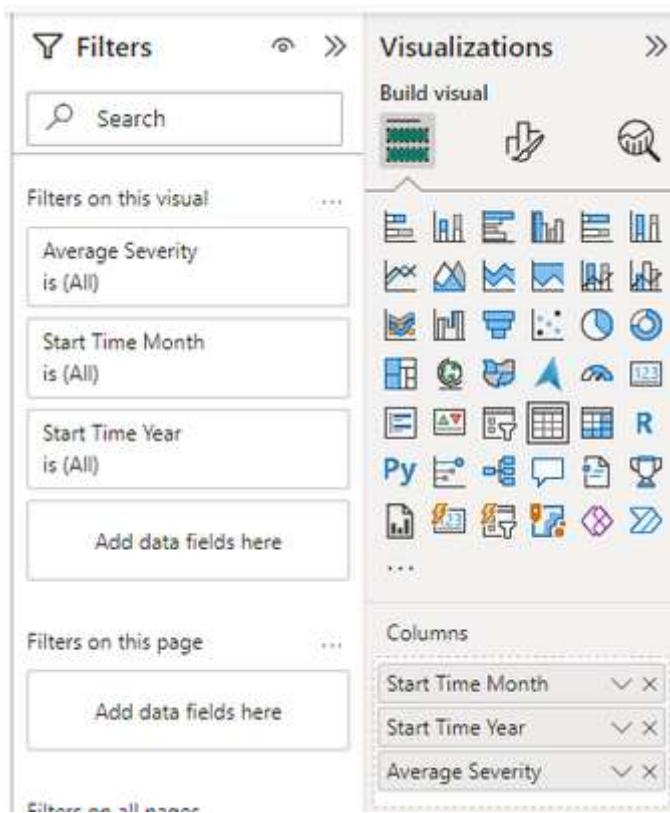
- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

Average Severity	Column Labels	1	2	3	11	12	Grand Total
Row Labels							
2022					3,480573694	3,470139973	3,474403211
2023		3,573114565	3,279396815	3,191324201			3,452739726
Grand Total		3,573114565	3,279396815	3,191324201	3,480573694	3,470139973	3,464932251

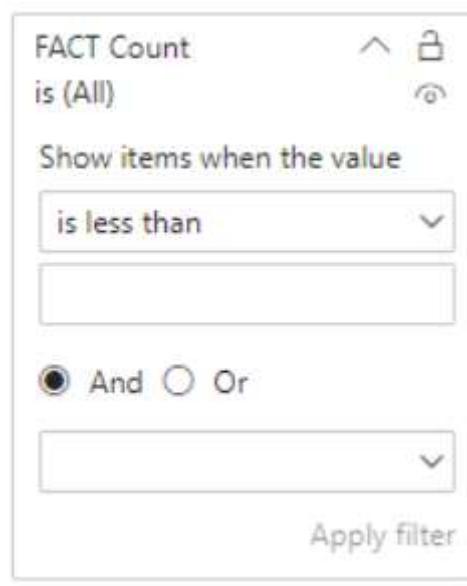
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

### 6.5.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị



- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.



## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- Kết quả:

Start Time Month	Start Time Year	Average Severity
11	2022	3.48
12	2022	3.47
1	2023	3.57
2	2023	3.28
3	2023	3.19
<b>Total</b>		<b>3.46</b>

### 6.6. Hiển thị tổng khoảng cách và số vụ tai nạn xảy ra từng tháng của từng năm với loại thời tiết là FAIR.

#### 6.6.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

Start Time Year	Start Time Month	FACT Count	Sum Distance米
2022	11	42880	42380.9989999999
2022	12	46068	42542.169
2023	1	40871	30575.736
2023	2	17105	18906.349
2023	3	7606	6032.1039999999

#### 6.6.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT NON EMPTY {[Dim Date].[Start Time Year].children * [Dim Date].[Start Time Month].children} ON ROWS,
       {[Measures].[FACT Count], [Measures].[Sum Distance米]} ON COLUMNS
FROM [Accidents DW]
WHERE ([Dim Weather].[Weather Condition].[Fair])
```

- Kết quả:

## *Đồ án xây dựng kho dữ liệu US ACCIDENTS*

		FACT Count	Sum Distance mi
2022	11	42880	42380.99899999999
2022	12	46068	42542.169
2023	1	40871	30575.736
2023	2	17105	18906.349
2023	3	7606	6032.10399999999

### **6.6.3. Thực hiện trên Excel.**

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotChart Fields

Choose fields to add to report:

Search 

**Average Severity**

FACT Count

**Sum Distancemi**

Dim Airport

Airport Code

Airport ID

Drag fields between areas below:

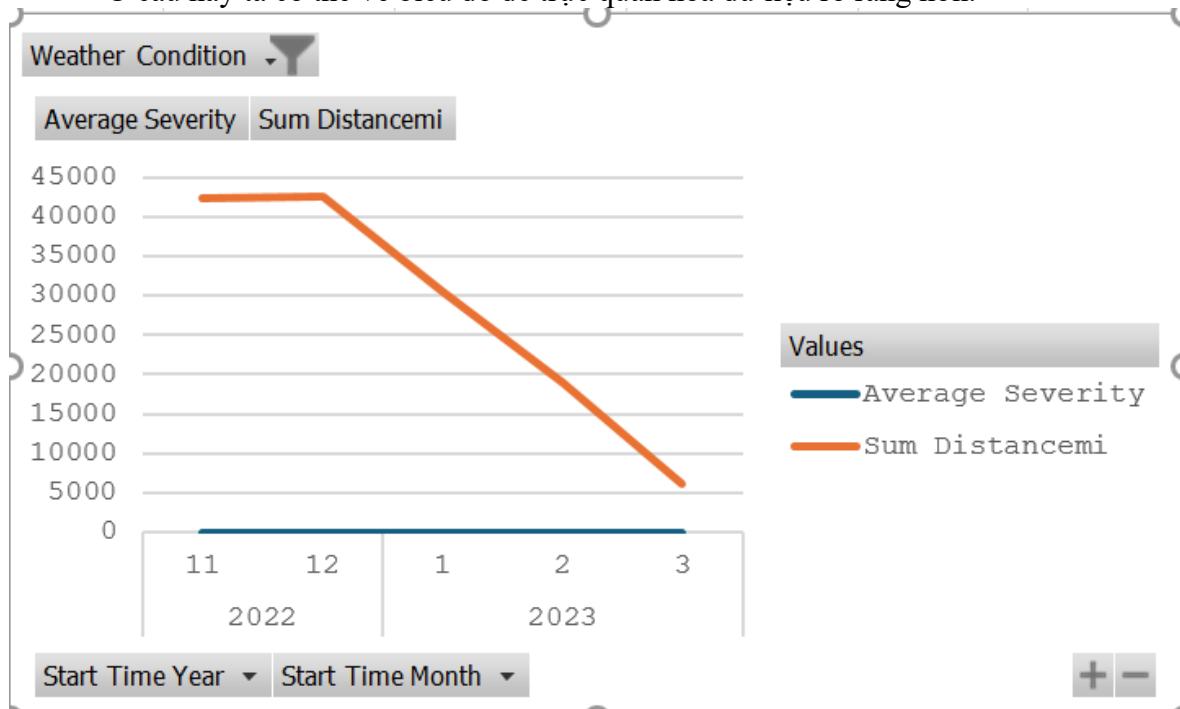
FILTERS	LEGEND (SERIES)
Weather Condition	$\Sigma$ Values
AXIS (CATEGORIES)	$\Sigma$ VALUES
Start Time Year	Average Severity
Start Time Month	Sum Distancemi

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

A	B	C
Weather Condition	Fair	
Row Labels	Average Severity	Sum Distancemi
2022		
11	3,144875317	42380,999
12	3,051456593	42542,169
2023		
1	3,168904058	30575,736
2	3,008243393	18906,349
3	2,902907296	6032,104
<b>Grand Total</b>	<b>3,09427675</b>	<b>140437,357</b>

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.



### 6.6.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

The screenshot shows the 'Filters' pane in Power BI. It includes sections for 'Filters on this visual', 'Filters on this page', and 'Filters on all pages'. Under 'Filters on this visual', there are four items: 'Average Severity is (All)', 'Start Time Month is (All)', 'Start Time Year is (All)', and 'Sum Distance is (All)'. Under 'Weather Condition', it says 'is Fair'. Below these is a section for 'Add data fields here'. In the 'Rows' section, 'Start Time Month' and 'Weather Condition' are listed. In the 'Columns' section, 'Start Time Year' is listed. In the 'Values' section, 'Average Severity' and 'Sum Distance' are listed. A 'Build visual' section is also visible.

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.

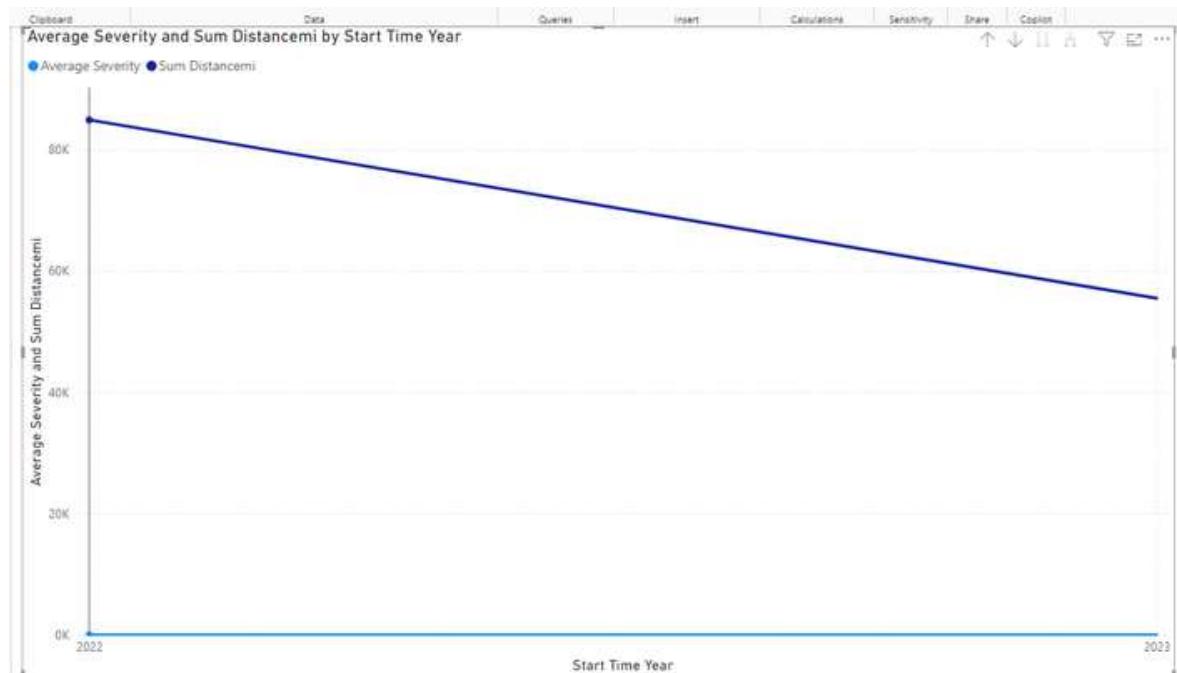
The screenshot shows the filter dialog for 'FACT Count'. It starts with 'FACT Count is (All)'. Below that is a section titled 'Show items when the value' with a dropdown set to 'is less than'. There is a text input field below the dropdown. At the bottom, there are two radio buttons: 'And' (selected) and 'Or'. Below them is another dropdown menu. At the very bottom is a button labeled 'Apply filter'.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- Kết quả:

Start Time Year	2022		2023		Total		
	Start Time Month	Average Severity	Sum Distancemi	Average Severity	Sum Distancemi	Average Severity	Sum Distancemi
1				3.17	30,575.74	3.17	30,575.74
2				3.01	18,906.35	3.01	18,906.35
3				2.90	6,032.10	2.90	6,032.10
11		3.14	42,381.00			3.14	42,381.00
12		3.05	42,542.17			3.05	42,542.17
Total		3.10	84,923.17	3.09	55,514.19	3.09	140,437.36

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.



## 6.7. Liệt kê các quận và quận có số vụ tai nạn từ 3000 trở lên.

### 6.7.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

The screenshot shows the SSAS MDX Query Editor interface. On the left, the 'Metadata' pane displays the schema of the 'Accidents DW' database, including dimensions like Dim Location, Dim Date, and Dim Airport, and measures like FACT Count. A filter expression is applied to the Dim Location dimension to select counties where the fact count is greater than or equal to 3000. The main pane shows a table of county names and their corresponding fact counts.

County	FACT Count
Alameda	4487
Dallas	5781
Denton	3054
Harris	4561
Hennepin	3216
Los Angeles	25151
Marcopas	5121
Medfordburg	3009
Miami-Dade	13381
Montgomery	3630
Orange	5460
Orange	4644
Riverside	4924
Sacramento	5198
San Bernardino	6677
San Diego	6429
Santa Clara	3346
Travis	5499
Wake	4747

### 6.7.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT [Measures].[Fact Count] ON COLUMNS,
NON EMPTY {[Dim Location].[County].[County].MEMBERS}
HAVING [Measures].[Fact Count] >= 3000 ON ROWS
FROM [Accidents DW]
```

- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

FACT Count	
Alameda	4487
Dallas	5781
Davidson	3054
Harris	4561
Hennepin	3216
Los Angeles	25151
Maricopa	5131
Mecklenburg	3609
Miami-Dade	12381
Montgomery	3620
Orange	5400
Orange	4644
Riverside	4924
Sacramento	5198
San Bernardino	6677
San Diego	6429
Santa Clara	3346
Travis	5499
Wake	4747

### 6.7.3. Thực hiện trên Excel.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotChart Fields

Choose fields to add to report:

Search 

FACT Count

Average Severity

Sum Distance

Dim Airport

Airport Code

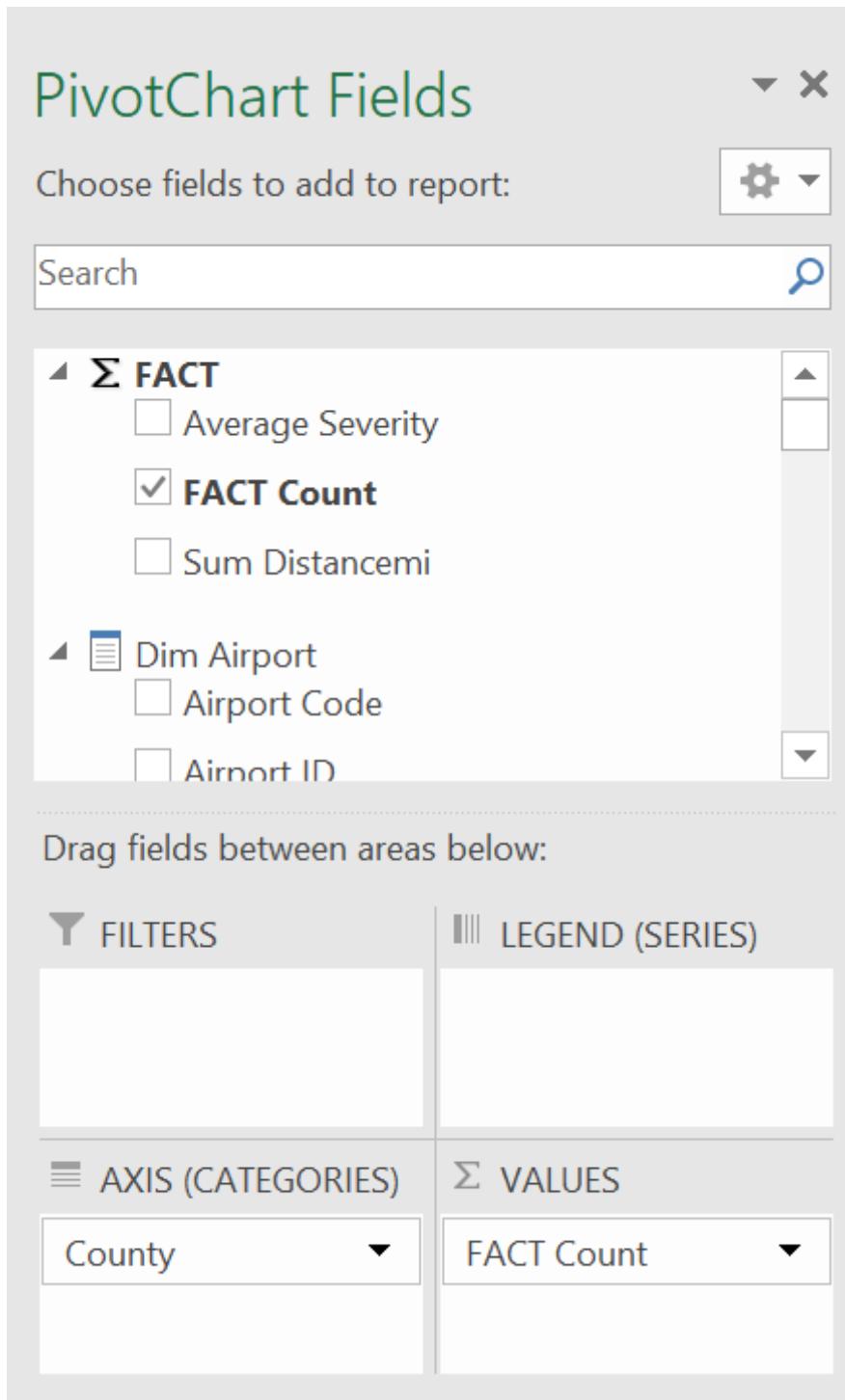
Airnort ID

Drag fields between areas below:

 FILTERS  LEGEND (SERIES)

 AXIS (CATEGORIES)  VALUES

County ▼ FACT Count ▼



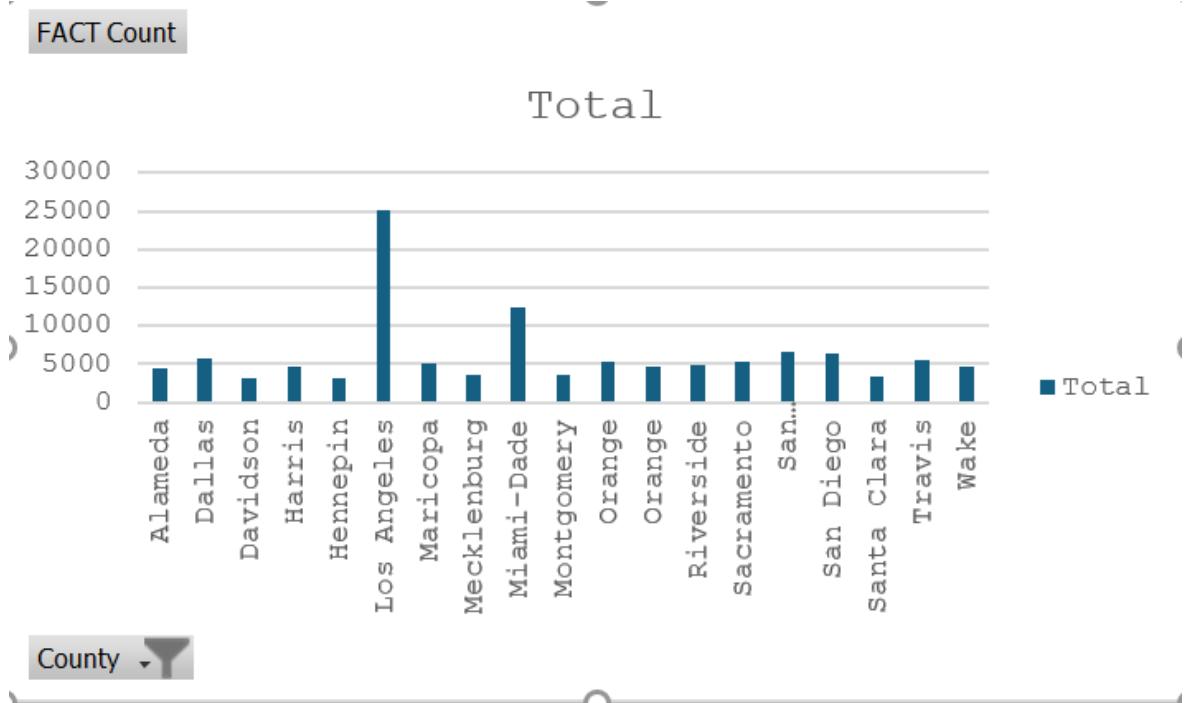
- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Row Labels	FACT Count
Alameda	4487
Dallas	5781
Davidson	3054
Harris	4561
Hennepin	3216
Los Angeles	25151
Maricopa	5131
Mecklenburg	3609
Miami-Dade	12381
Montgomery	3620
Orange	5400
Orange	4644
Riverside	4924
Sacramento	5198
San Bernardino	6677
San Diego	6429
Santa Clara	3346
Travis	5499
Wake	4747
<b>Grand Total</b>	<b>117855</b>

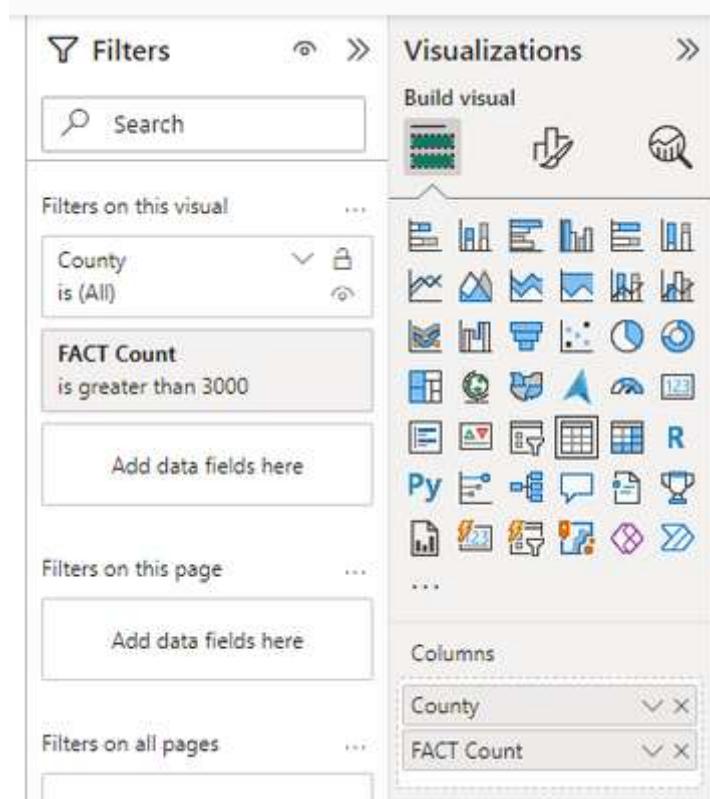
- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



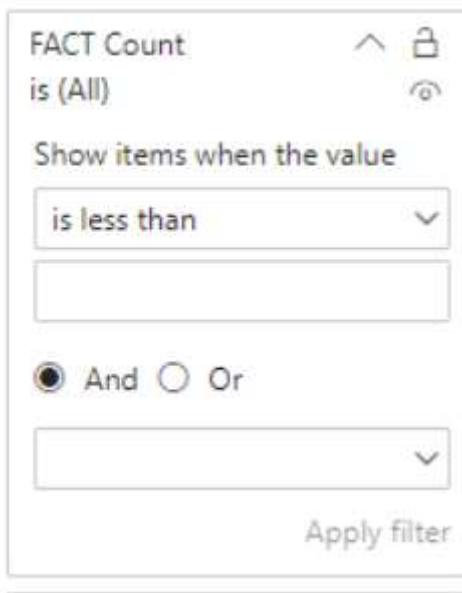
### 6.7.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị



## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.



- Kết quả:

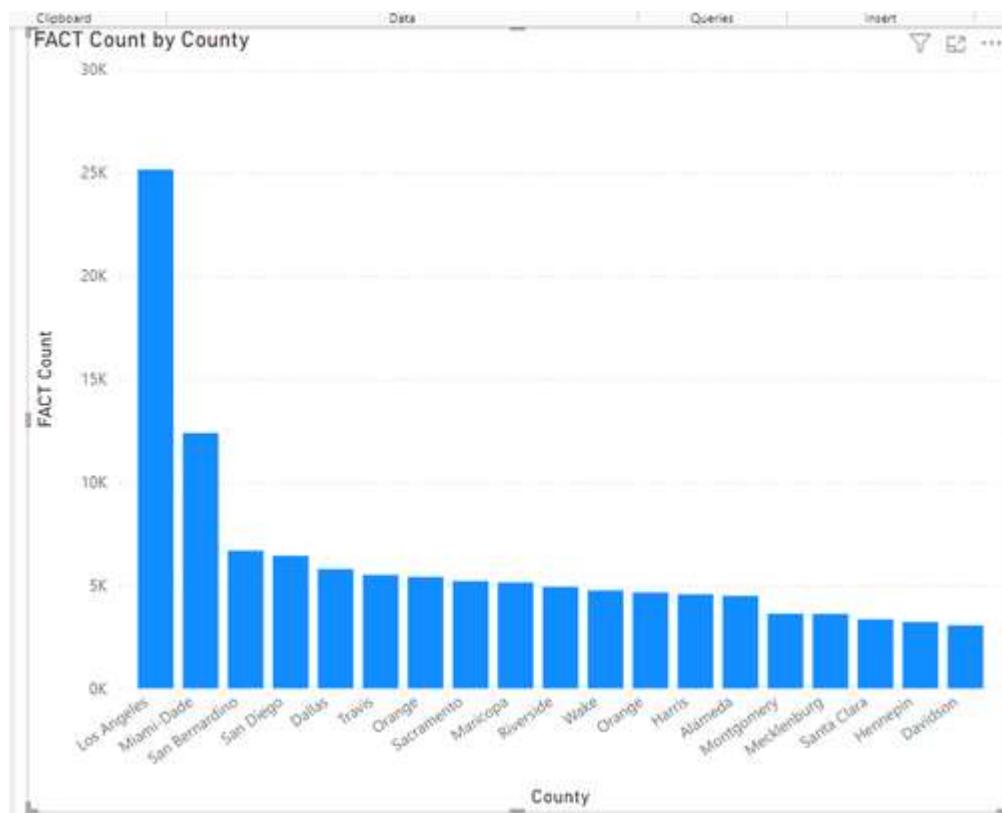
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

The screenshot shows a Microsoft Excel spreadsheet with a table titled "Clipboard". The table has two columns: "County" and "FACT Count". The data includes various counties and their corresponding fact counts, ending with a total row. The "FACT Count" column uses a color scheme where most values are in blue, except for the total which is in orange.

County	FACT Count
Alameda	4487
Dallas	5781
Davidson	3054
Harris	4561
Hennepin	3216
Los Angeles	25151
Maricopa	5131
Mecklenburg	3609
Miami-Dade	12381
Montgomery	3620
Orange	5400
Orange	4644
Riverside	4924
Sacramento	5198
San Bernardino	6677
San Diego	6429
Santa Clara	3346
Travis	5499
Wake	4747
<b>Total</b>	<b>117855</b>

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.

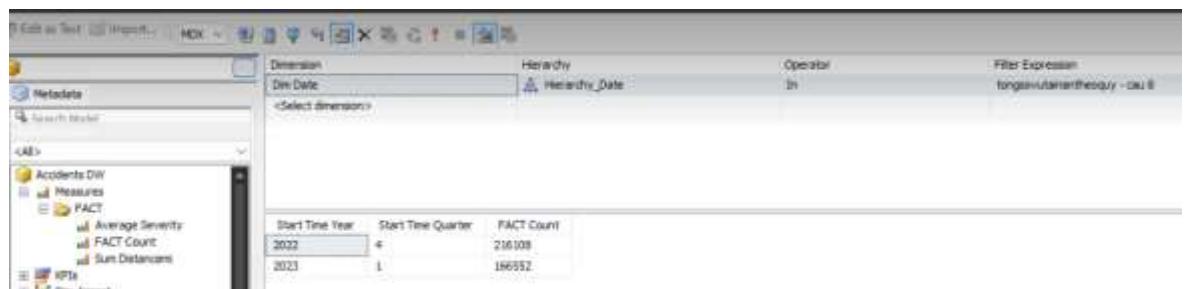
## Đồ án xây dựng kho dữ liệu US ACCIDENTS



## 6.8. Tổng số vụ tai nạn theo quý của từng năm (DRILLDOWNLEVEL).

### 6.8.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.



### 6.8.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT DRILLDOWNLEVEL([Dim Date].[Hierarchy_Date].[Start Time YEAR]) ON ROWS,  
      NON EMPTY [Measures].[Fact Count] ON COLUMNS  
FROM [Accidents DW]
```

## *Đồ án xây dựng kho dữ liệu US ACCIDENTS*

- Kết quả:

	FACT Count
2022	216108
4	216108
2023	166552
1	166552
Unknown	(null)
Unknown	(null)

### **6.8.3. Thực hiện trên Excel.**

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotTable Fields

Choose fields to add to report:

Search 

▲ **Σ FACT**

- Average Severity
- FACT Count**
- Sum Distance

▲  **Dim Airport**

- Airport Code
- Airnort ID

Drag fields between areas below:

FILTERS	COLUMNS

ROWS	Σ VALUES
Start Time Quarter ▾	FACT Count ▾

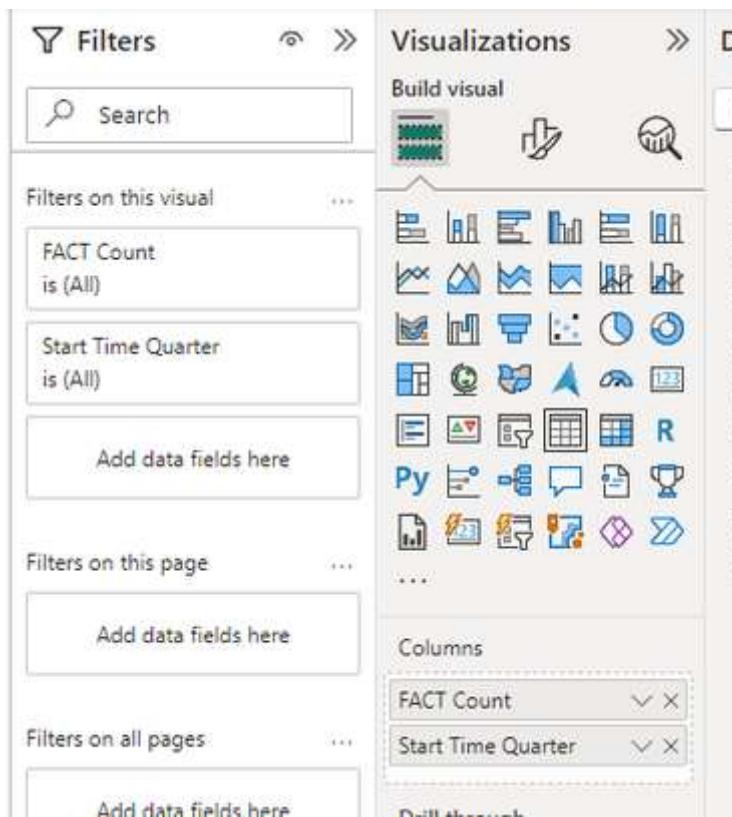
- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Row Labels	FACT Count
1	166552
4	216108
<b>Grand Total</b>	<b>382660</b>

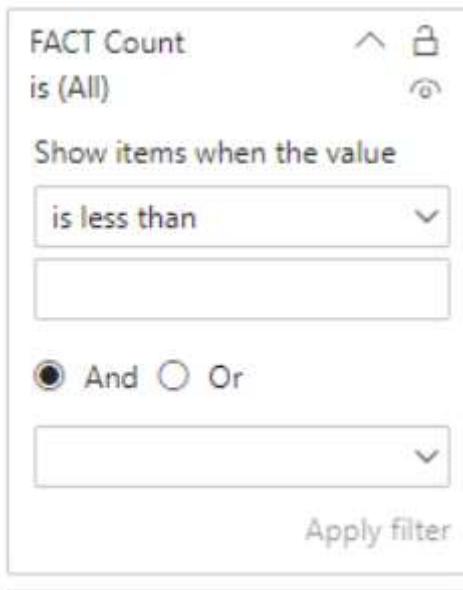
### 6.8.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị



- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



- Kết quả:

FACT Count	Start Time	Quarter
166552	1	
216108	4	
<b>382660</b>		

### 6.9. Top 4 tháng của 2 năm có số lượng tai nạn cao nhất theo bang (Hàm GENERATE với TOP COUNT).

#### 6.9.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Start Time Month	State	FACT Count
1	AL	610
1	AR	765
1	AZ	2379
1	CA	24278
1	CO	1539
1	CT	943
1	DC	470
1	DE	106
1	FL	13877
1	GA	2890
1	IA	269
1	ID	97
1	IL	1474
1	IN	598
1	KS	312
1	KY	51
1	LA	2056
1	MA	75
1	MD	2152
1	MI	1877
1	MN	4699
1	MO	1141
1	MS	196
1	MT	1142
1	NC	4336
1	ND	162
1	NE	33
1	NJ	1438
1	NM	68
1	NV	446
1	NY	3658
1	OH	3446
1	OK	419
1	OR	1887
1	PA	4877

### 6.9.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

```
SELECT {[Measures].[Fact Count]} * [Dim Location].[State].children} ON COLUMNS,  
    GENERATE([Dim Location].[State],  
        TOPCOUNT([Dim Date].[Start Time Month].children, 4, [Measures].[Fact Count])) ON ROWS  
FROM [Accidents DW]
```

- Kết quả:

Messages		Results																			
FACT Count																					
AI	AR	AZ	CA	CD	CT	DC	DE	FL	GA	IA	ID	IL	IN	IS	KY	LA	MD	MI	MS	NC	NC
12	833	732	2710	26311	1701	1074	546	125	14355	3235	564	121	1888	1976	498	65	21				
1	610	765	2379	24276	1538	343	470	106	13877	2656	283	57	1474	598	312	31	20				
11	541	557	2834	18607	1055	1134	499	106	19123	1362	261	115	1208	676	360	71	14				
2	1	invR	1218	18702	967	912	1	invR	391	1774	42	61	963	289	237	1	9				

### 6.9.3. Thực hiện trên Excel.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

PivotChart Fields

Choose fields to add to report:

Search 

▲ **Σ FACT**

- Average Severity
- FACT Count
- Sum Distance

▲ **Dim Airport**

- Airport Code
- Airnort ID

Drag fields between areas below:

FILTERS	LEGEND (SERIES)
	State

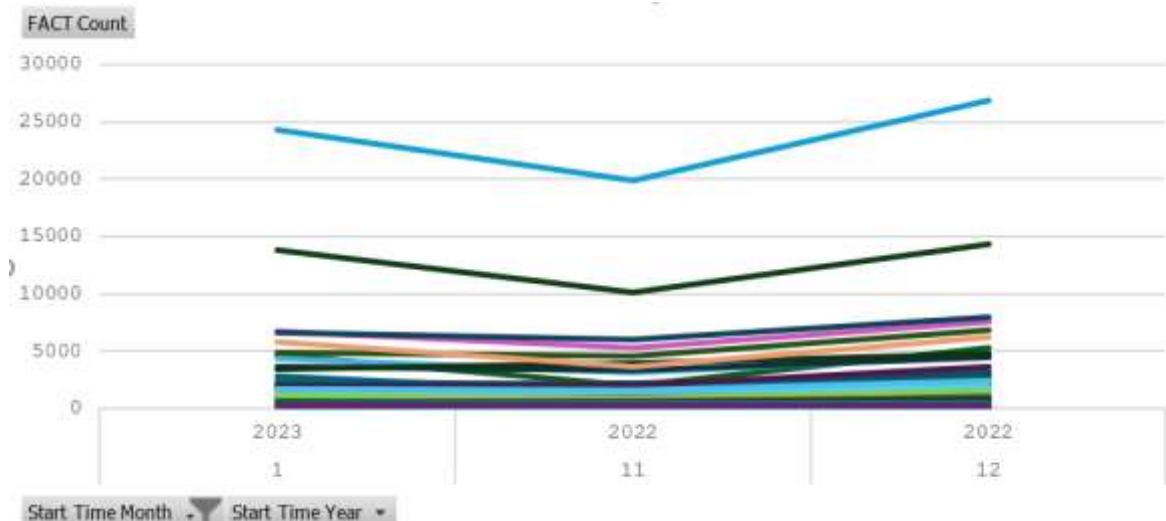
AXIS (CATEGORIES)	Σ VALUES
Start Time Month	FACT Count
Start Time Year	

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

FACT Count	Column Labels	AR	AZ	CA	CO	CT	DC	DE	FL	GA	IA	ID	IL	IN	KS	KY	LA	MA	MD	MI
=1																				
	2023	610	765	2379	24278	1539	943	470	106	13877	2890	269	97	1474	598	312	51	2056	75	2152
=11																				
	2022	541	597	2034	19927	1055	1134	488	106	10123	1362	201	115	1208	676	360	71	1460	61	2187
=12																				
	2022	683	732	2710	26911	1701	1074	546	126	14350	3235	504	121	1680	1076	498	65	2122	62	3073
	Grand Total	1834	2094	7123	71116	4295	3151	1504	338	38350	7487	974	333	4362	2350	1170	187	5638	198	7412
																			535	

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.



### 6.9.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

The screenshot shows the 'Filters' pane in Power BI. It includes sections for 'Filters on this visual', 'Filters on this page', and 'Filters on all pages'. Under 'Filters on this visual', there are filters for 'FACT Count' (is (All)), 'Start Time Month' (top 3 by Count of Star...), and 'State' (is (All)). There is also a button 'Add data fields here'. The 'Visualizations' pane on the right shows a grid of icons for different chart types like bar charts, line charts, and maps. Below the visualization pane are sections for 'X-axis' (set to 'Start Time Month'), 'Y-axis' (set to 'FACT Count'), 'Secondary y-axis' (button 'Add data fields here'), and 'Legend' (set to 'State').

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.

The screenshot shows the filter settings for 'FACT Count' (is (All)). It includes a dropdown 'Show items when the value' set to 'is less than', a dropdown for the value (empty), and radio buttons for 'And' (selected) and 'Or'. At the bottom is a button 'Apply filter'.

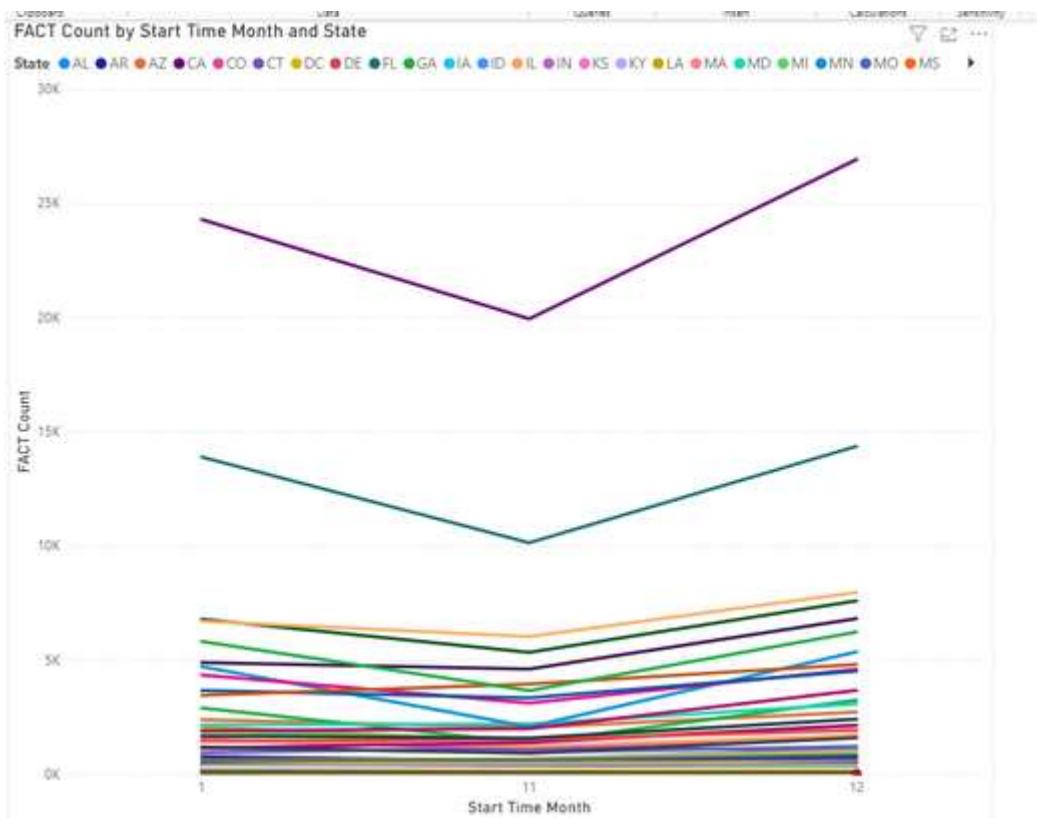
## Đồ án xây dựng kho dữ liệu US ACCIDENTS

- Kết quả:

Clipboard	Data
Start Time Month	FACT Count State
1	610 AL
1	765 AR
1	2379 AZ
1	24278 CA
1	1539 CO
1	943 CT
1	470 DC
1	106 DE
1	13877 FL
1	2890 GA
1	269 IA
1	97 ID
1	1474 IL
1	598 IN
1	312 KS
1	51 KY
1	2056 LA
1	75 MA
1	2152 MD
1	1877 MI
1	4699 MN
1	1141 MO
1	196 MS
1	1142 MT
1	4336 NC
1	162 ND
1	33 NE
1	1438 NJ
1	68 NM
1	446 NV
1	3658 NY
1	3446 OH
1	419 OK
Total	324923

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



### 6.10. Số vụ tai nạn theo từng năm của các tiểu bang ngoài trừ bang NY và bang NV.

#### 6.10.1. Thực hiện trên các khối Cubes.

- Chúng ta kéo thả các thuộc tính và measure phù hợp với câu truy vấn ta đang làm.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Dimension	Hierarchy	Operator	Filter Expression
Dim Location	State	Not Equal	{NY, NV}
State	Start Time Year	FACT Count	
KY	2022	136	
KY	2023	52	
LA	2022	3582	
LA	2023	2056	
MA	2022	123	
MA	2023	77	
MD	2022	5260	
MD	2023	2152	
MI	2022	3474	
MI	2023	3356	
MN	2022	7417	
MN	2023	6075	
MO	2022	2153	
MO	2023	1568	
MS	2022	276	
MS	2023	196	
MT	2022	3514	
MT	2023	1415	
NC	2022	7692	
NC	2023	6149	
ND	2022	247	
ND	2023	162	
NE	2022	141	
NE	2023	65	
NJ	2022	3418	
NJ	2023	3188	
NM	2022	117	
NM	2023	68	
NH	2022	1	
OH	2022	8734	
OH	2023	3449	

### 6.10.2. Thực hiện trên SQL.

- Ta thực hiện câu truy vấn bằng SQL.

```
SELECT [Measures].[Fact Count] ON COLUMNS,  
NON EMPTY {EXCEPT([Dim Location].[State].CHILDREN,  
{[Dim Location].[State].&[NY],[Dim Location].[State].&[NV]}) * [Dim Date].[Start Time Year].[Start Time Year]} ON ROWS  
FROM [Accidents DW]
```

- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

FACT Count		
MI	2023	3356
MN	2022	7417
MN	2023	6075
MO	2022	2153
MO	2023	1568
MS	2022	276
MS	2023	196
MT	2022	3514
MT	2023	1415
NC	2022	7692
NC	2023	6149
ND	2022	247
ND	2023	162
NE	2022	141
NE	2023	65
NJ	2022	3418
NJ	2023	3188
NM	2022	117
NM	2023	68
NH	2022	1
OH	2022	8734
OH	2023	3449
OK	2022	788
OK	2023	419
OR	2022	5648
OR	2023	3990
PA	2022	11397
PA	2023	7120
RI	2022	86
RI	2023	34

### 6.10.3. Thực hiện trên Excel.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

### PivotChart Fields

Choose fields to add to report:

Search 

FACT Count

Average Severity

Sum Distance

Dim Airport

Airport Code

Airnort ID

Drag fields between areas below:

FILTERS	LEGEND (SERIES)

AXIS (CATEGORIES)	VALUES
State	FACT Count
Start Time Year	

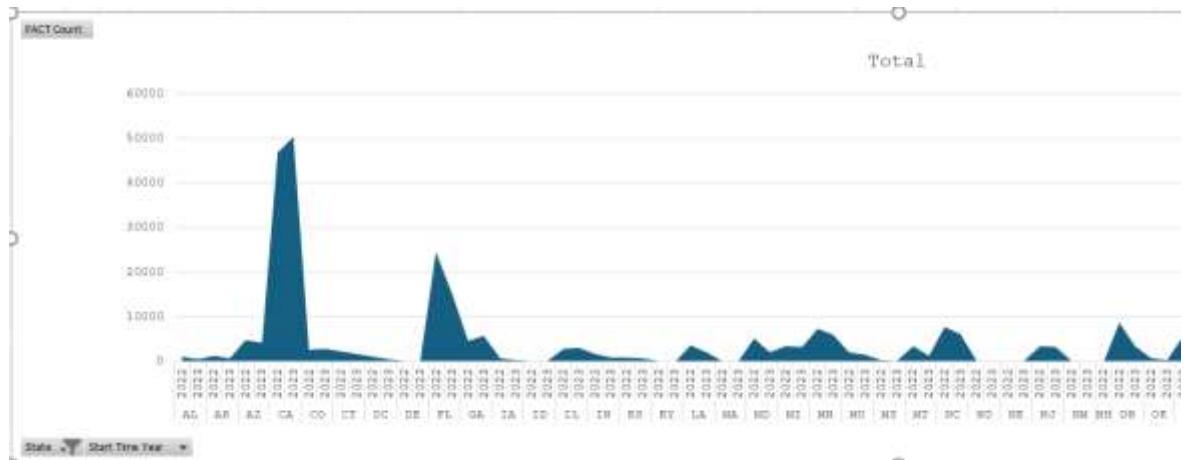
- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng  của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Row Labels	FACT Count
AL	
2022	1224
2023	611
AR	
2022	1329
2023	765
AZ	
2022	4744
2023	4180
CA	
2022	46838
2023	50287
CO	
2022	2756
2023	2882
CT	
2022	2208
2023	1712
DC	
2022	1034
2023	471
DE	
2022	232
2023	106
FL	
2022	24473
2023	14959
GA	
2022	4597

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



### 6.10.4. Thực hiện trên PowerBI.

- Ta thực hiện kéo thả các thuộc tính ta cần truy vấn vào đúng ô giá trị

The screenshot shows the PowerBI interface's 'Visualizations' pane. It includes sections for 'Filters', 'Visualizations', 'Data', and 'Build visual'. Under 'Build visual', there are numerous icons representing different types of charts and reports. On the right, the 'X-axis' and 'Y-axis' settings are displayed, both currently set to 'State' and 'FACT Count' respectively.

- Ta thực hiện các điều kiện của câu truy vấn tại biểu tượng của cột hoặc hàng tùy theo điều kiện của truy vấn ta làm.

## *Đồ án xây dựng kho dữ liệu US ACCIDENTS*

FACT Count ^  

is (All) 

Show items when the value

is less than ▼

And  Or ▼

**Apply filter**

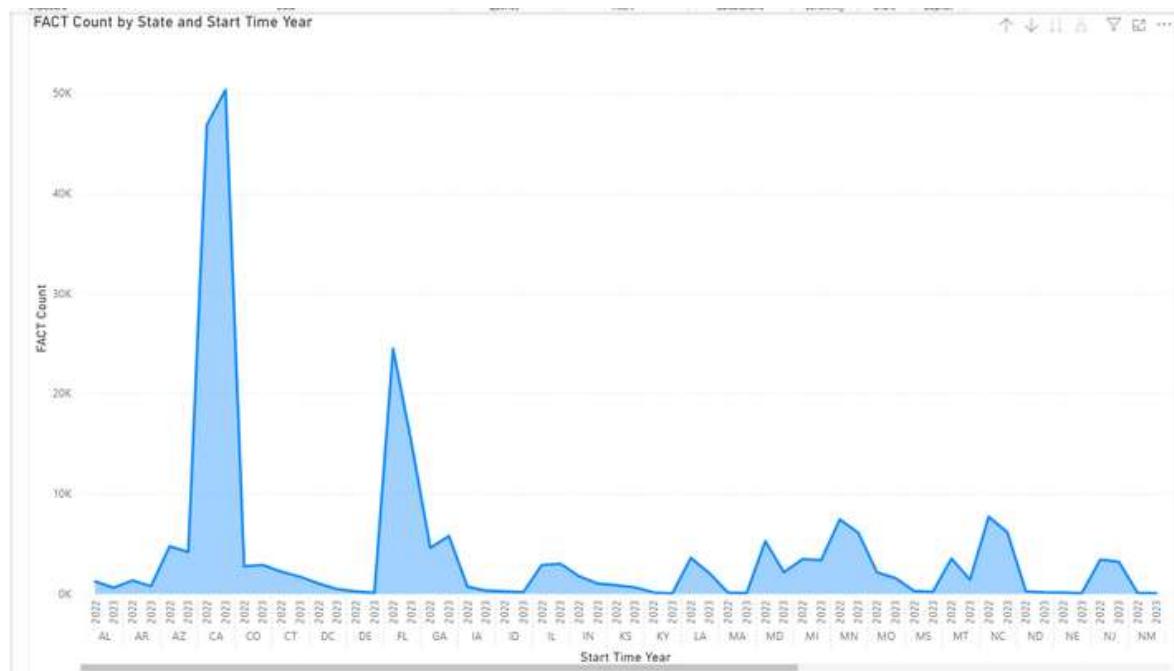
- Kết quả:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

State	Start Time	Year	FACT Count
AL		2022	1224
AL		2023	611
AR		2022	1329
AR		2023	765
AZ		2022	4744
AZ		2023	4180
CA		2022	46838
CA		2023	50287
CO		2022	2756
CO		2023	2882
CT		2022	2208
CT		2023	1712
DC		2022	1034
DC		2023	471
DE		2022	232
DE		2023	106
FL		2022	24473
FL		2023	14959
GA		2022	4597
GA		2023	5774
IA		2022	705
IA		2023	322
ID		2022	236
ID		2023	178
IL		2022	2888
IL		2023	2994
IN		2022	1752
IN		2023	1015
KS		2022	858
KS		2023	643
KY		2022	136
KY		2023	52
LA		2022	3582
<b>Total</b>			<b>366144</b>

- Ở câu này ta có thể vẽ biểu đồ để trực quan hóa dữ liệu rõ ràng hơn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



## CHƯƠNG 4. QUÁ TRÌNH DATAMINING

Chủ đề: Dự đoán mức độ nghiêm trọng (Severity) của tai nạn giao thông ở Mỹ.

### 4.1. Tiền xử lý dữ liệu

#### 4.1.1. Bổ sung tính năng

Chúng tôi quyết định phân tách tính năng Start\_Time thành các thành phần năm, tháng, ngày, giờ và phút để đưa chúng vào mô hình.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```

df_new['Start_Time'] = df_new['Start_Time'].str.split('.').str[0]
df_new['Start_Time'] = pd.to_datetime(df_new['Start_Time'])
# Extract year, month, day, hour and weekday
df_new['Year'] = df_new['Start_Time'].dt.year
df_new['Month'] = df_new['Start_Time'].dt.month
df_new['Weekday'] = df_new['Start_Time'].dt.weekday
df_new['Day'] = df_new['Start_Time'].dt.day

df_new['hour'] = df_new['Start_Time'].dt.hour
df_new['minute'] = df_new['Start_Time'].dt.minute

df_new.head()

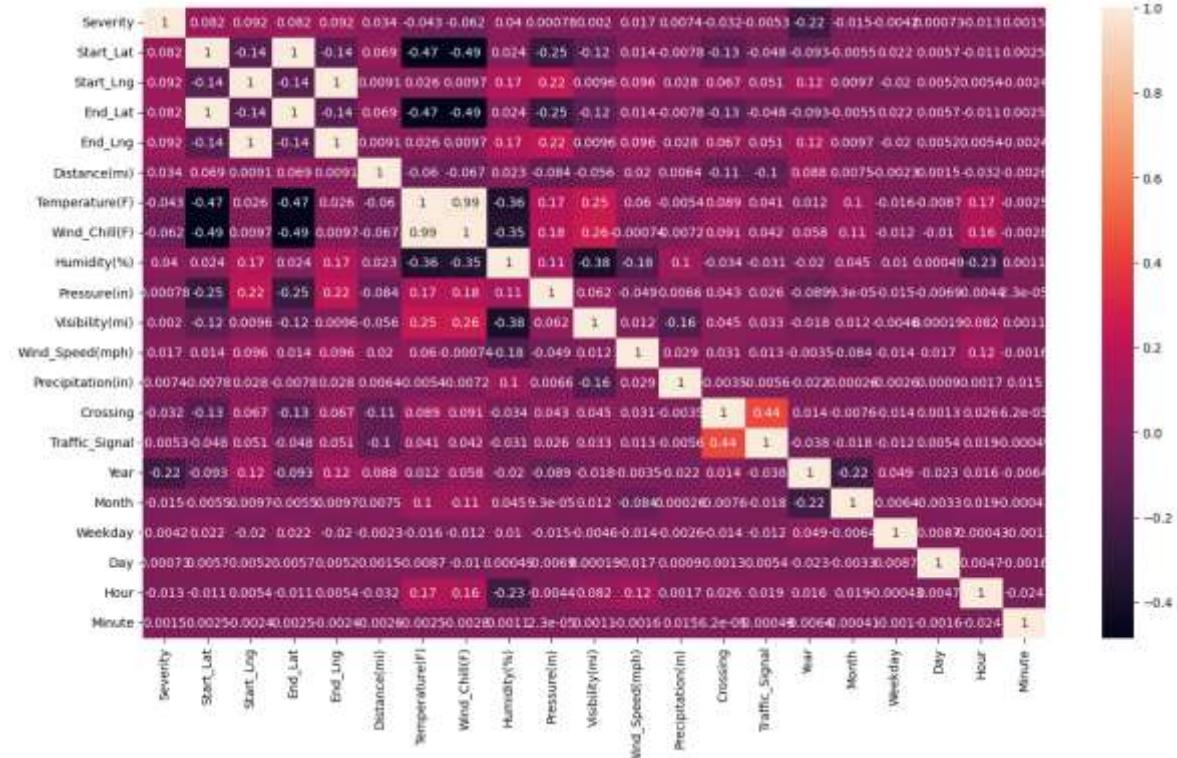
```

ID	Source	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Sunrise_Sunset	Civil_Twilight	Nautical_Twilight	Astr
4000000	A- SourceI	3	2022-12-13 12:45:19	2022-12-13 14:02:12	43.045713	-88.047363	43.038323	-88.047223	0.510	—	Day	Day	Day
4000001	A- SourceI	2	2022-01-17 12:38:00	2022-01-17 14:09:30	29.783896	-95.546130	29.783931	-95.543861	0.136	—	Day	Day	Day
4000002	A- SourceI	3	2022-11-08 07:03:30	2022-11-08 08:49:30	38.748783	-76.879469	38.768111	-76.884740	1.363	—	Day	Day	Day
4000003	A- SourceI	2	2022-11-01 04:42:00	2022-11-01 04:05:06	34.109993	-118.010919	34.108345	-118.010237	0.120	—	Night	Night	Night
4000004	A- SourceI	3	2022-12-19 10:48:00	2022-12-19 12:15:42	42.357973	-79.531784	42.382474	-79.488299	2.792	—	Day	Day	Day

5 rows × 14 columns

### 4.1.2. Kiểm tra mối tương quan giữa các tính năng

Trong phần tiếp theo, chúng tôi trình bày ma trận tương quan giữa tất cả các tính năng có thể có, dưới dạng biểu đồ nhiệt. Từ đó, chúng ta có thể quan sát mối tương quan giữa các tính năng khác nhau của tập dữ liệu, để kiểm tra xem một số tính năng có tương quan cao hay không và loại bỏ một trong số chúng.



Từ ma trận này, chúng ta có thể thấy rằng tọa độ GPS bắt đầu (Start\_Lat, Start\_Lng) và kết thúc (End\_Lat, End\_Lng) của các vụ tai nạn có mối tương quan cao. Hơn nữa, chỉ số Wind\_Chill (temperature) tỷ lệ thuận với nhiệt độ, vì

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

vậy chúng ta có thể loại bỏ một trong hai tính năng này.

Chúng ta cũng có thể thấy rằng sự có mặt của tín hiệu giao thông (traffic signal) có mối tương quan nhỏ với mức độ nghiêm trọng của một vụ tai nạn, có nghĩa là có thể đèn giao thông có thể giúp cho giao thông thông suốt khi xảy ra tai nạn. Từ ma trận, chúng ta cũng có thể thấy rằng chúng ta không thể tính được hiệp phương sai với Turning\_Loop, và điều đó là do nó luôn luôn là False.

### 4.1.3. Lựa chọn tính năng

Dưới đây là quá trình lựa chọn tính năng, nhằm chọn ra các tính năng tốt nhất để mô hình của chúng ta có thể học từ đó.

Từ các quan sát được thực hiện với ma trận tương quan, chúng tôi sẽ loại bỏ các tính năng sau: End\_Lat, End\_Lng, Wind Chill. Ngoài ra, chúng tôi cũng sẽ loại bỏ các tính năng sau:

- ID: vì chúng không mang bất kỳ thông tin nào về mức độ nghiêm trọng của tai nạn
- Start\_Time: vì nó đã được phân tách thành các tính năng thời gian được thêm trước đó (ngày, tháng, ngày trong tuần)
- End\_Time: vì chúng ta không thể biết trước khi nào lưu lượng giao thông trở lại bình thường
- Description: hầu hết các mô tả chỉ báo tên đường của tai nạn, vì vậy chúng tôi quyết định bỏ qua tính năng này để đơn giản hóa
- Number, Street, County, State, Zipcode, Country: vì chúng ta chỉ tập trung vào Thành phố nơi xảy ra tai nạn
- Timezone, Airport\_Code, Weather\_Timestamp: vì chúng không hữu ích cho nhiệm vụ của chúng ta
- Turning\_Loop: vì nó luôn luôn là False
- Sunrise\_Sunset, Nautical\_Twilight, Astronomical\_Twilight: vì chúng là trùng lặp.

```
features_to_drop2 = ['ID', 'source', 'start_time', 'end_time', 'Description', 'Weather_Timestamp', 'Traffic_Calming', 'Street', 'County', 'State', 'Zipcode', 'Country', 'timezone', 'Airport_X']  
x = df_new.drop(features_to_drop2, axis=1)  
x.head()
```

	Severity	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	City	Temperature(F)	Wind_Chill(F)	Humidity(%)	...	Junction	Station	Stop	Traffic_Signal	Year
0	2	40.045713	-88.047363	40.033825	-88.047373	0.510	Milwaukee	36.0	28.0	70.0	—	False	False	False	True	2002
1	2	29.783896	-95.540130	29.783931	-95.543861	0.136	Houston	64.0	64.0	25.0	—	False	False	False	False	2022
2	2	39.748783	-76.679469	39.766111	-76.684740	1.365	Clinton	45.0	39.0	58.0	—	False	False	False	False	2002
3	2	34.100993	-118.010919	34.108345	-118.010237	0.120	Arcadia	66.0	66.0	84.0	—	False	False	False	False	2002
4	2	42.357973	-79.531704	42.382474	-79.488299	2.792	Westfield	21.0	21.0	69.0	—	False	False	False	False	2022

5 rows × 17 columns

### 4.1.4. Xóa trùng lặp

Trong phần này, chúng tôi sẽ kiểm tra xem liệu có các trùng lặp trong tập dữ liệu hay không.

```
] x.drop_duplicates(inplace=True)  
print(len(x.index))  
  
3375940
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

### 4.1.5. Xử lý giá trị sai hoặc thiếu sót

Ở đây, chúng tôi sẽ làm sạch tập dữ liệu từ các giá trị sai hoặc thiếu sót. Hãy bắt đầu từ cột Side:

Chúng ta có thể thấy rằng có một bản ghi không có Side, vì vậy chúng ta có thể bỏ nó.

Hãy phân tích Áp suất (Pressure) và Tầm nhìn (Visibility):

```
x[['Pressure(in)', 'Visibility(mi)']].describe().round(2)
```

	Pressure(in)	Visibility(mi)
count	3303403.00	3288483.00
mean	29.39	9.08
std	1.12	2.65
min	0.00	0.00
25%	29.22	10.00
50%	29.76	10.00
75%	29.99	10.00
max	58.63	140.00

Chúng ta có thể thấy rằng giá trị tối thiểu (Min) là 0, có nghĩa là một số bản ghi thiếu chúng và được thay thế bằng số 0. Vì lý do này, chúng tôi sẽ loại bỏ các bản ghi với giá trị thiếu cho hai cột này.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
X = X[X['Pressure(in)'] != 0]
X = X[X['Visibility(mi)'] != 0]
X[['Pressure(in)', 'Visibility(mi)']].describe().round(2)
```

	Pressure(in)	Visibility(mi)
count	3299134.00	3284206.00
mean	29.39	9.09
std	1.12	2.64
min	0.30	0.06
25%	29.22	10.00
50%	29.76	10.00
75%	29.99	10.00
max	58.63	140.00

Nếu chúng ta phân tích các điều kiện thời tiết (weather conditions), chúng ta có thể thấy rằng có rất nhiều điều kiện thời tiết, vì vậy tốt hơn là giảm số lượng điều kiện độc nhất.

```
weather_unique = X['Weather_Condition'].unique()
print(weather_unique)
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
[ 'Cloudy' 'Fair' 'Partly Cloudy' 'Light Snow' 'Mostly Cloudy' 'Light Rain'  
'Haze' 'Light Rain with Thunder' 'T-Storm' 'Fog' 'N/A Precipitation'  
'Rain' 'Cloudy / Windy' 'Freezing Drizzle' 'Light Freezing Drizzle'  
'Heavy Rain' 'Thunder' 'Light Snow / Windy' 'Shallow Fog' 'Light Drizzle'  
'Fair / Windy' 'Heavy T-Storm' 'Light Rain / Windy'  
'Blowing Snow / Windy' 'Mostly Cloudy / Windy' 'Wintry Mix' 'Snow'  
'Thunder in the Vicinity' 'Heavy Snow' 'Partly Cloudy / Windy'  
'Light Snow and Sleet / Windy' 'Light Freezing Rain' 'Rain / Windy'  
'Patches of Fog' 'Snow and Sleet / Windy' 'Haze / Windy' 'Blowing Snow'  
'Fog / Windy' 'Heavy Rain / Windy' 'T-Storm / Windy' 'Smoke'  
'Blowing Dust' 'Snow / Windy' 'Wintry Mix / Windy' 'Heavy Snow / Windy'  
'Light Rain Shower' 'Snow and Sleet' 'Sleet' 'Light Sleet' 'Drizzle'  
'Sleet / Windy' 'Heavy T-Storm / Windy' 'Light Snow Shower'  
'Light Snow and Sleet' 'Showers in the Vicinity' 'Thunder / Windy'  
'Light Rain Shower / Windy' 'Drizzle and Fog' 'Mist' 'Snow and Thunder'  
'Hail' 'Freezing Rain' 'Shallow Fog / Windy' 'Drizzle / Windy'  
'Thunder and Hail' 'Light Drizzle / Windy' 'Blowing Dust / Windy'  
'Mist / Windy' 'Light Freezing Rain / Windy' 'Widespread Dust'  
'Widespread Dust / Windy' 'Heavy Drizzle' 'Funnel Cloud'  
'Drifting Snow / Windy' 'Tornado' 'Freezing Rain / Windy' 'Heavy Sleet'  
'Squalls / Windy' 'Light Snow with Thunder' 'Light Sleet / Windy'  
'Rain Shower' 'Sleet and Thunder' 'Heavy Freezing Drizzle'  
'Smoke / Windy' 'Thunder and Hail / Windy' 'Heavy Sleet and Thunder'  
'Small Hail' 'Squalls' 'Thunder / Wintry Mix' 'Overcast'  
'Light Snow Shower / Windy' 'Sand / Dust Whirlwinds'  
'Heavy Sleet / Windy' 'Sand / Dust Whirls Nearby'  
'Heavy Snow with Thunder' 'Sand / Windy' 'Duststorm'  
'Heavy Rain Shower / Windy' 'Snow and Thunder / Windy'  
'Sand / Dust Whirlwinds / Windy' 'Blowing Snow Nearby' 'Snow Grains'  
'Partial Fog' 'Heavy Freezing Rain' 'Blowing Sand'  
'Thunder / Wintry Mix / Windy' 'Patches of Fog / Windy'  
'Heavy Rain Shower' 'Drifting Snow' 'Scattered Clouds' 'Clear'  
'Light Freezing Fog' 'Thunderstorm' 'Light Thunderstorms and Snow'  
'Heavy Thunderstorms and Rain' 'Ice Pellets' 'Thunderstorms and Rain'  
'Heavy Blowing Snow' 'Light Rain Showers' 'Light Thunderstorms and Rain'  
'Light Ice Pellets' 'Rain Showers' 'Low Drifting Snow'  
'Light Blowing Snow' 'Heavy Rain Showers' 'Light Snow Showers'  
'Light Haze' 'Heavy Thunderstorms with Small Hail' 'Sand']
```

Để làm điều này, chúng tôi sẽ thay thế chúng bằng mô tả chung hơn:

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
x.loc[X['Weather_Condition'].str.contains('Thunder|T-Storm',na=False),'Weather_Condition'] = 'Thunderstorm'
x.loc[X['Weather_Condition'].str.contains('Snow|Sleet|Wintry',na=False),'Weather_Condition'] = 'Snow'
x.loc[X['Weather_Condition'].str.contains('Rain|Drizzle|Shower',na=False),'Weather_Condition'] = 'Rain'
x.loc[X['Weather_Condition'].str.contains('Wind|Squalls',na=False),'Weather_Condition'] = 'Windy'
x.loc[X['Weather_Condition'].str.contains('Hail|Pellets',na=False),'Weather_Condition'] = 'Hail'
x.loc[X['Weather_Condition'].str.contains('Fair',na=False),'Weather_Condition'] = 'Clear'
x.loc[X['Weather_Condition'].str.contains('Cloud|Overcast',na=False),'Weather_Condition'] = 'Cloudy'
x.loc[X['Weather_Condition'].str.contains('Mist|Haze|Fog',na=False),'Weather_Condition'] = 'Fog'
x.loc[X['Weather_Condition'].str.contains('Sand|Dust',na=False),'Weather_Condition'] = 'Sand'
x.loc[X['Weather_Condition'].str.contains('Smoke|Volcanic Ash',na=False),'Weather_Condition'] = 'Smoke'
x.loc[X['Weather_Condition'].str.contains('N/A Precipitation',na=False),'Weather_Condition'] = numpy.nan
print(X['Weather_Condition'].unique())

['Cloudy' 'Clear' 'Snow' 'Rain' 'Fog' 'Thunderstorm' nan 'Windy' 'Smoke'
 'Sand' 'Hail' 'Tornado']
```

Hãy kiểm tra cả trường hợp Hướng gió (Wind\_Direction):

```
| x.loc[X['Wind_Direction'] == 'CALM','Wind_Direction'] = 'Calm'
| x.loc[X['Wind_Direction'] == 'VAR','Wind_Direction'] = 'Variable'
| x.loc[X['Wind_Direction'] == 'East','Wind_Direction'] = 'E'
| x.loc[X['Wind_Direction'] == 'North','Wind_Direction'] = 'N'
| x.loc[X['Wind_Direction'] == 'South','Wind_Direction'] = 'S'
| x.loc[X['Wind_Direction'] == 'West','Wind_Direction'] = 'W'
| X['Wind_Direction'] = X['Wind_Direction'].map(lambda x : x if len(x) != 3 else x[1:], na_action = 'ignore')
| X['Wind_Direction'].unique()

array(['SE', 'Calm', 'N', 'NW', 'SW', 'S', 'W', 'NE', 'E', nan,
       'Variable'], dtype=object)
```

Như chúng ta có thể thấy, chúng ta có thể nhóm các giá trị như chúng ta đã làm với Điều kiện thời tiết (weather conditions):

Tiếp theo, hãy phân tích các giá trị bị thiếu:

```
X.isna().sum()
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
Severity          0
Start_Lat         0
Start_Lng         0
End_Lat          0
End_Lng          0
Distance(mi)      0
City              147
Temperature(F)    83956
Wind_Chill(F)     308520
Humidity(%)       88836
Pressure(in)      72528
Visibility(mi)     87456
Wind_Direction    97133
Wind_Speed(mph)   146751
Precipitation(in) 382739
Weather_Condition 87547
Crossing          0
Junction          0
Station            0
Stop               0
Traffic_Signal    0
Year               0
Month              0
Weekday            0
Day                0
Hour               0
Minute             0
dtype: int64
```

Vì rất nhiều bản ghi không có thông tin về Lượng mưa (Precipitation), chúng tôi sẽ loại bỏ tính năng này. Đối với các tính năng số, chúng tôi sẽ điền giá trị bị thiếu bằng giá trị trung bình, trong khi đối với các tính năng phân loại như Thành phố (City), Hướng gió (Wind\_Direction), Điều kiện thời tiết (Weather\_Condition) và Chạng vạng dân sự (Civil\_Twilight), chúng tôi sẽ xóa các bản ghi với thông tin bị thiếu.

```
# Remove the 'Precipitation(in)' column
features_to_fill = ['Temperature(F)', 'Wind_Chill(F)', 'Humidity(%)', 'Pressure(in)', 'Visibility(mi)', 'Wind_Speed(mph)', 'Precipitation(in)']
X[features_to_fill] = X[features_to_fill].fillna(X[features_to_fill].mean())
X.dropna(inplace=True)
X.isna().sum()
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
Severity          0  
Start_Lat        0  
Start_Lng        0  
End_Lat          0  
End_Lng          0  
Distance(mi)     0  
City              0  
Temperature(F)   0  
Wind_Chill(F)    0  
Humidity(%)      0  
Pressure(in)     0  
Visibility(mi)   0  
Wind_Direction   0  
Wind_Speed(mph)  0  
Precipitation(in) 0  
Weather_Condition 0  
Crossing          0  
Junction          0  
Station            0  
Stop               0  
Traffic_Signal    0  
Year               0  
Month              0  
Weekday            0  
Day                0  
Hour               0  
Minute             0  
dtype: int64
```

### 4.1.6. Kiểm tra phương sai của tính năng

Trong phần này, chúng tôi sẽ kiểm tra phương sai cho mỗi tính năng để loại bỏ các tính năng có phương sai rất thấp vì chúng không thể giúp phân biệt các trường hợp.

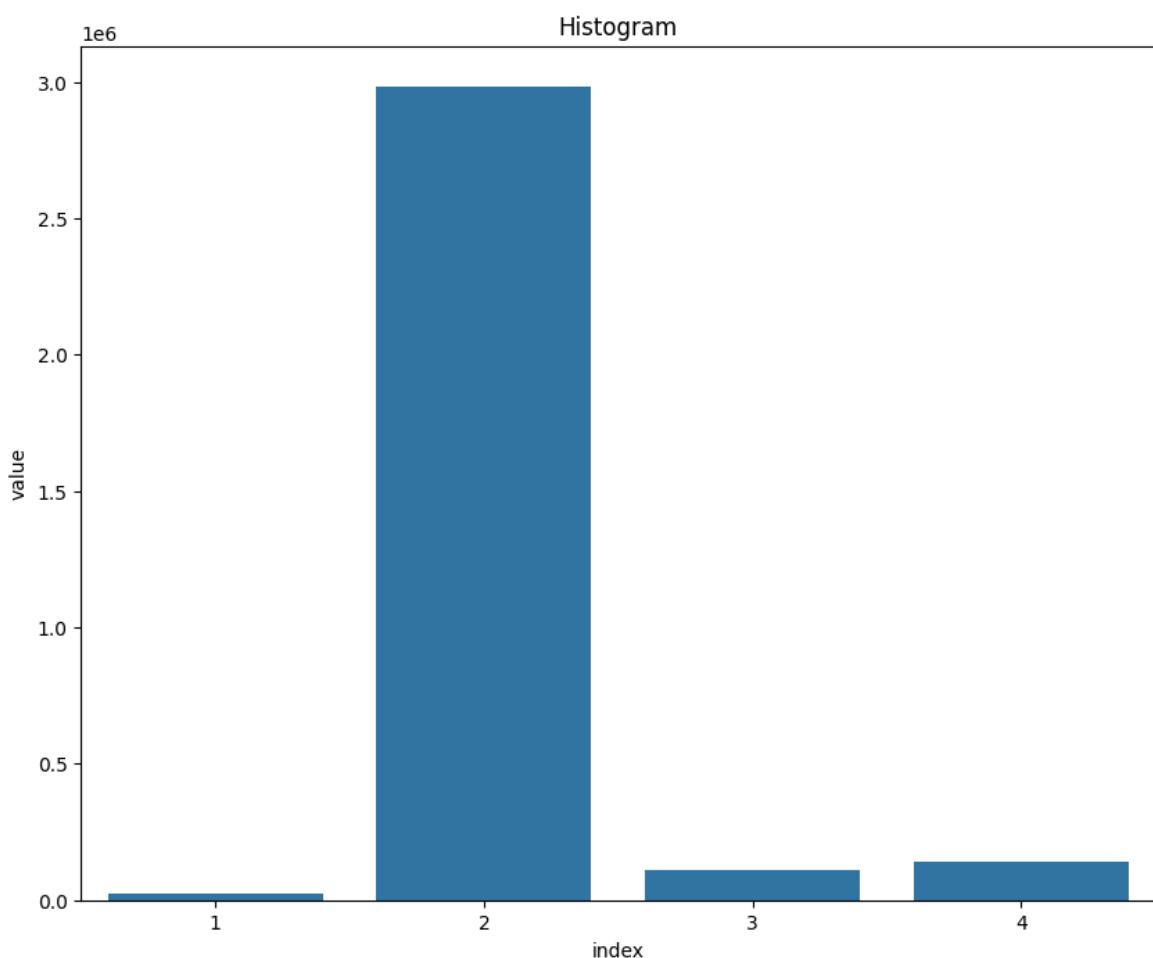
X.describe().T													
	Severity	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Temperature(F)	Wind_Chill(F)	Humidity(%)	Pressure(in)	Visibility(mi)	Wind_Speed(mph)	Precipita
count	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00	3259653.00
mean	2.11	36.26	-96.10	36.26	-96.10	0.84	61.06	59.42	63.77	29.40	9.09	7.44	%
std	0.45	5.37	10.13	5.27	10.12	1.83	19.31	21.19	22.98	1.11	2.63	5.91	
min	1.00	24.57	-124.55	24.57	-124.55	0.00	-69.00	-89.00	1.00	0.30	0.08	0.00	
25%	2.00	33.46	-117.84	33.46	-117.84	0.06	48.00	48.00	47.00	29.22	10.05	3.00	
50%	2.00	36.14	-89.97	36.14	-89.97	0.28	63.00	61.00	66.00	29.76	10.05	7.00	
75%	2.00	40.14	-60.32	40.14	-60.32	0.91	76.00	75.00	83.00	35.99	10.05	10.00	
max	4.00	49.00	-67.48	49.50	-67.48	155.19	207.00	207.00	100.00	59.63	140.05	1087.00	

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

Mặc dù Lượng mưa (Precipitation) và Áp suất (Pressure) có phương sai nhỏ, nhưng không cần phải loại bỏ chúng vì chúng thường có giá trị nhỏ.

### 4.1.7. Xử lý dữ liệu không cân bằng

```
severity_counts = X['Severity'].value_counts()  
plt.figure(figsize=(10,8))  
plt.title('Histogram')  
sns.barplot(x = severity_counts.index, y = severity_counts.values)  
plt.xlabel('index')  
plt.ylabel('value')
```



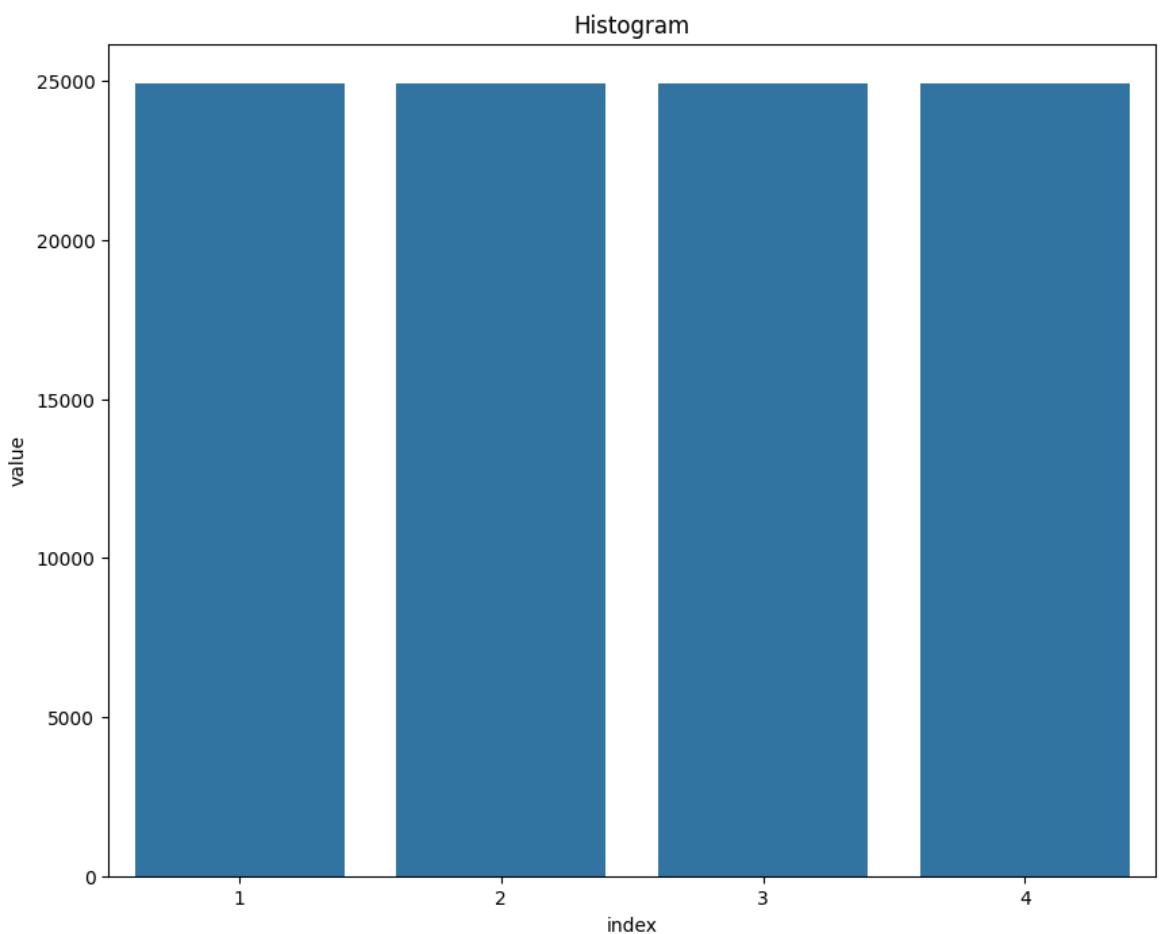
Thuộc tính nghiêm trọng (severity) như chúng ta đã thấy từ biểu đồ trước đó là rất không cân bằng, số vụ tai nạn có mức độ nghiêm trọng 1 rất nhỏ trong khi số vụ tai nạn có mức độ nghiêm trọng 2 cao hơn nhiều.

Do đó, để cân bằng dữ liệu, chúng tôi sẽ giảm số lượng bản ghi của tất cả các danh mục để bằng với số lượng bản ghi của danh mục thiểu số, trong trường hợp này là mức độ nghiêm trọng 1. Chúng tôi nghĩ rằng đây là một lựa chọn tốt vì điều này sẽ giúp chúng tôi có một số lượng bản ghi tốt cho mỗi danh mục. Đó là ~ 15 nghìn bản ghi

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
size = len(X[X['Severity'] == 1].index)
df_f = pd.DataFrame()
for i in range(1,5):
    S = X[X['Severity'] == i]
    df_f = pd.concat([df_f, S.sample(size,random_state=42)])
X = df_f
```

```
severity_counts = X['Severity'].value_counts()
plt.figure(figsize=(10,8))
plt.title('Histogram')
sns.barplot(x = severity_counts.index, y = severity_counts.values)
plt.xlabel('index')
plt.ylabel('value')
```



### 4.1.8. Chuẩn hóa tính năng

Trong phần này, chúng tôi sẽ chuẩn hóa và điều chỉnh các tính năng. Để cài

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

thiên hiệu suất của các mô hình của chúng tôi, chúng tôi đã chuẩn hóa các giá trị của các tính năng liên tục.

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
features_scaler = ['WindHility(mi)', 'Wind_Speed(mph)', 'Start_Lat', 'Start_Lng', 'Distance(mi)', 'Temperature(F)', 'Wind_Chill(F)', 'Humidity(%)', 'Precipitation(in)', 'Year', 'Month']
X[features_scaler] = scaler.fit_transform(X[features_scaler])
X.head()
```

	Severity	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	City	Temperature(F)	Wind_Chill(F)	Humidity(%)	...	Junction	Station	Stop	Traffic_Signal	Y
2977500	1	32.219838	-80.615907	29.945056	-90.13184	0.0	New Orleans	0.477733	0.527299	0.758102	—	False	False	False	False	False
3006665	1	33.368937	-112.020199	33.58185	-112.18604	0.0	Glendale	0.582096	0.622572	0.051020	—	False	False	False	True	
3012594	1	38.573029	-121.051731	38.57561	-121.57537	0.0	West Sacramento	0.562753	0.604251	0.285714	—	False	False	False	True	
3042734	1	36.047241	-117.489184	35.49313	-117.19460	0.0	Harm	0.461538	0.512642	0.763300	—	False	False	False	True	
2977467	1	36.047172	-117.59449	36.24077	-116.79551	0.0	Nashville	0.378518	0.435691	0.857143	—	False	False	False	False	

5 rows × 17 columns

### 4.1.9. Mã hóa tính năng

Cuối cùng, trong phần này, chúng tôi sẽ mã hóa các tính năng phân loại.

```
categorical_features = set(['City', 'Wind_Direction', 'Weather_Condition'])
for cat in categorical_features:
    X[cat] = X[cat].astype('category')
X.info()
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
<class 'pandas.core.frame.DataFrame'>
Index: 99712 entries, 2977500 to 1275932
Data columns (total 27 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Severity        99712 non-null   int64  
 1   Start_Lat       99712 non-null   float64 
 2   Start_Lng       99712 non-null   float64 
 3   End_Lat         99712 non-null   float64 
 4   End_Lng         99712 non-null   float64 
 5   Distance(mi)   99712 non-null   float64 
 6   City            99712 non-null   category
 7   Temperature(F) 99712 non-null   float64 
 8   Wind_Chill(F)  99712 non-null   float64 
 9   Humidity(%)    99712 non-null   float64 
 10  Pressure(in)   99712 non-null   float64 
 11  Visibility(mi) 99712 non-null   float64 
 12  Wind_Direction 99712 non-null   category
 13  Wind_Speed(mph) 99712 non-null   float64 
 14  Precipitation(in) 99712 non-null   float64 
 15  Weather_Condition 99712 non-null   category
 16  Crossing        99712 non-null   bool   
 17  Junction         99712 non-null   bool   
 18  Station          99712 non-null   bool   
 19  Stop             99712 non-null   bool   
 20  Traffic_Signal  99712 non-null   bool   
 21  Year             99712 non-null   float64 
 22  Month            99712 non-null   float64 
 23  Weekday          99712 non-null   float64 
 24  Day              99712 non-null   float64 
 25  Hour             99712 non-null   float64 
 26  Minute           99712 non-null   float64 
dtypes: bool(5), category(3), float64(18), int64(1)
memory usage: 16.4 MB
```

Đầu tiên, chúng tôi sẽ hiển thị số lượng lớp độc nhất cho mỗi tính năng phân loại.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
print('Unique class for each category: ')
for cat in categorical_features:
    print('{:15s}'.format(cat), '\t', len(x[cat].unique()))
```

```
Unique class for each category:
Wind_Direction          10
Weather_Condition        10
City                      6995
```

Sau đó, chúng tôi sẽ mã hóa các giá trị Boolean dưới dạng số.

```
x = x.replace({True, False}, {1,0})
x.head()
```

C:\Users\thevl\AppData\Local\Temp\ipykernel\_6484\496001633.py:1: FutureWarning: Downcasting behavior in 'replace' is deprecated and will be removed in a future version. To retain:
x = x.replace({True, False}, {1,0})

	Severity	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	City	Temperature(F)	Wind_Chill(F)	Humidity(%)	...	Junction	Station	Stop	Traffic_Signal	W
2977500	1	0.210938	-0.615907	29.94506	-90.13184	0.0	New Orleans	0.477733	0.527299	0.755102	...	0	0	0	0	0
3006668	1	0.368937	0.220199	33.38185	-112.18604	0.0	Glendale	0.582996	0.622572	0.051020	...	0	0	0	0	1
3012594	1	0.573529	0.051731	38.57561	-121.57537	0.0	West Sacramento	0.562793	0.604281	0.285714	...	0	0	0	0	1
3042734	1	0.447241	0.489184	35.49313	-97.19480	0.0	Hanah	0.461538	0.512642	0.765306	...	0	0	0	0	1
2977467	1	0.477872	0.675949	36.24077	-86.78551	0.0	Nashville	0.376518	0.435691	0.857143	...	0	0	0	0	0

5 rows × 27 columns

Bây giờ chúng tôi có thể mã hóa các tính năng phân loại bằng phương pháp get\_dummies(), biến đổi các tính năng bằng mã hóa one-hot.

```
onehot_cols = list(categorical_features - set(['city']))
x = pd.get_dummies(x, columns=onehot_cols, drop_first=True)
x.head()
```

	Severity	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	City	Temperature(F)	Wind_Chill(F)	Humidity(%)	...	Wind_Direction_N	Weather_Condition_Cloudy	...	...	...	...
2977500	1	0.210938	-0.615907	29.94506	-90.13184	0.0	New Orleans	0.477733	0.527299	0.755102	...	False	True	...	...	...	...
3006668	1	0.368937	0.220199	33.38185	-112.18604	0.0	Glendale	0.582996	0.622572	0.051020	...	True	False	...	...	...	...
3012594	1	0.573529	0.051731	38.57561	-121.57537	0.0	West Sacramento	0.562793	0.604281	0.285714	...	False	False	...	...	...	...
3042734	1	0.447241	0.489184	35.49313	-97.19480	0.0	Hanah	0.461538	0.512642	0.765306	...	False	False	...	...	...	...
2977467	1	0.477872	0.675949	36.24077	-86.78551	0.0	Nashville	0.376518	0.435691	0.857143	...	False	False	...	...	...	...

5 rows × 43 columns

Bây giờ, chỉ còn lại để mã hóa tính năng Thành phố (City). Để giảm sử dụng bộ nhớ và số lượng tính năng, chúng tôi đã sử dụng BinaryEncoder được bao gồm trong thư viện category\_encoders.

```
import category_encoders as ce
binary_encoder = ce.binary.BinaryEncoder()
city_binary_enc = binary_encoder.fit_transform(x['city'])
city_binary_enc
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

	city_0	city_1	city_2	city_3	city_4	city_5	city_6	city_7	city_8	city_9	city_10	city_11	city_12
2977500	0	0	0	0	0	0	0	0	0	0	0	0	1
3006665	0	0	0	0	0	0	0	0	0	0	0	1	0
3012594	0	0	0	0	0	0	0	0	0	0	0	1	1
3042734	0	0	0	0	0	0	0	0	0	0	1	0	0
2977467	0	0	0	0	0	0	0	0	0	0	1	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2960012	0	0	1	1	0	1	0	0	0	1	0	0	1
1396157	0	0	0	0	0	0	0	1	1	0	1	1	1
3690592	0	0	1	0	1	1	1	1	0	1	1	1	1
1389205	0	1	1	0	0	1	0	1	0	1	1	0	1
1275932	0	0	0	0	0	0	0	1	1	1	0	0	1

99712 rows × 13 columns

Cuối cùng, chúng tôi có thể kết hợp hai bảng dữ liệu và thu được bảng dữ liệu cuối cùng X với các tính năng phân loại được mã hóa.

```
X = pd.concat([X, city_binary_enc], axis = 1).drop('City', axis=1)
X.head()
```

	Severity	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Temperature(F)	Wind_Chill(F)	Humidity(%)	Pressure(in)	...	City_3	City_4	City_5	City_6	City_7
2977500	1	0.219938	0.619907	-29.94306	-90.13184	0.0	0.477733	0.327299	0.755102	30.10	—	0	0	0	0	0
3006665	1	0.368937	0.220199	33.58185	-112.18664	0.0	0.582996	0.622572	0.051030	28.48	—	0	0	0	0	0
3012594	1	0.973529	0.051731	38.87561	-121.57537	0.0	0.562731	0.604251	0.285714	29.72	—	0	0	0	0	0
3042734	1	0.447241	0.489164	35.40113	-97.19480	0.0	0.461538	0.512642	0.765006	28.77	—	0	0	0	0	0
2977467	1	0.477872	0.67949	-36.24077	-98.78551	0.0	0.276518	0.432691	0.857143	29.58	—	0	0	0	0	0

5 rows × 55 columns

## 4.2. Ứng dụng mô hình thuật toán khai thác dữ liệu

Khai báo 6 từ điển (dictionary) trống, mỗi từ điển có tên là accuracy, precision, recall, f1, fpr và tpr.

- accuracy: Lưu trữ kết quả độ chính xác (accuracy) của mô hình trên từng lớp.
- precision: Lưu trữ kết quả độ chính xác dự báo (precision) của mô hình trên từng lớp.
- recall: Lưu trữ kết quả độ phủ sóng (recall) của mô hình trên từng lớp.
- f1: Lưu trữ kết quả độ F1 (F1-score) của mô hình trên từng lớp.
- fpr: Lưu trữ tỷ lệ dự báo sai (false positive rate) của mô hình trên từng lớp.
- tpr: Lưu trữ tỷ lệ dự báo thật (true positive rate) của mô hình trên từng lớp.

Mỗi từ điển sẽ được sử dụng để lưu kết quả đánh giá cho một lớp (class) trong bài toán phân loại đa lớp (multi-class classification). Các giá trị sẽ được cập nhật trong quá trình đánh giá mô hình.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
accuracy = dict()
precision = dict()
recall = dict()
f1 = dict()
fpr = dict()
tpr = dict()
```

### 4.2.1. Chia dữ liệu trước khi xây dựng mô hình thuật toán

Sau khi tiền xử lý dữ liệu, chúng ta thu được DataFrame cuối cùng X với các thuộc tính phân loại được mã hóa.

DataFrame X được chia thành các tập lần lượt Train-Validate-Test

**Bước 1:** Chia DataFrame X thành hai phần: tập huấn luyện/đánh giá (X) và tập kiểm tra (X\_test).

```
] X, X_test = train_test_split(X,test_size=.2, random_state=42)
print(X.shape, X_test.shape)

(79769, 55) (19943, 55)
```

Sử dụng hàm train\_test\_split() từ thư viện scikit-learn để thực hiện việc chia. Thiết lập tham số test\_size bằng 0.2, có nghĩa là 20% dữ liệu sẽ được dùng cho tập kiểm tra và 80% còn lại sẽ được sử dụng cho huấn luyện và đánh giá.

- Thiết lập tham số random\_state bằng 42 để đảm bảo việc chia là có thể tái lập.
- In ra kích thước của hai tập kết quả sử dụng thuộc tính shape của mỗi DataFrame, trả về một tuple chứa số hàng và số cột (theo thứ tự đó).

**Bước 2:** Gán DataFrame X vào một biến mới là sample. Chia DataFrame sample thành hai phần: tập huấn luyện (X\_train và y\_train) và tập đánh giá (X\_validate và y\_validate), sử dụng hàm train\_test\_split() từ thư viện scikit-learn.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
sample = X
y_sample = sample['Severity']
X_sample = sample.drop('Severity',axis=1)

X_train, X_validate, y_train, y_validate = train_test_split (X_sample, y_sample, random_state=42)
print(X_train.shape, y_train.shape)
print(X_validate.shape, y_validate.shape)

(59826, 54) (59826,)
(19943, 54) (19943,)
```

- Trước hết, gán DataFrame X vào một biến mới là sample.
- Trích xuất thuộc tính mục tiêu Severity từ sample và gán cho y\_sample, còn lại các thuộc tính được gán cho X\_sample.
  - Sử dụng hàm train\_test\_split() để chia X\_sample và y\_sample thành tập huấn luyện và tập đánh giá. Thiết lập tham số random\_state bằng 42 để đảm bảo việc chia là có thể tái lập.
  - Cuối cùng, in ra kích thước của hai tập kết quả sử dụng thuộc tính shape của mỗi DataFrame hoặc Series.

### 4.2.2. Decision Tree

#### Xây dựng và dự đoán với mô hình Decision Tree

##### Bước 1.

- Tạo mô hình cây quyết định (dtc) sử dụng DecisionTreeClassifier từ sklearn.tree, với tham số Random\_state được đặt thành 42.
- Tạo danh sách các tham số để thử trong tìm kiếm dạng lười. Danh sách này bao gồm hai tham số: criterion (gồm hai phương pháp gini index và entropy đo độ tạp chất của nút) và max\_depth (độ sâu tối đa của cây).
- Tạo đối tượng tìm kiếm dạng lười với mô hình cây quyết định ‘dtc’, các tham số để thử ‘parameters’, ‘verbose = 5’ để in kết quả tìm kiếm dạng lười và n\_jobs = -1 để sử dụng tất cả các lõi CPU có sẵn.
- Huấn luyện mô hình trên tập huấn luyện ‘X\_train’ và ‘y\_train’, đồng thời thực hiện tìm kiếm dạng lười để tìm các tham số tốt nhất cho mô hình. Sau đó in ra các tham số tốt nhất, điểm số trên tập huấn luyện và điểm số trên tập hợp lệ của mô hình tốt nhất được tìm thấy bằng tìm kiếm lười.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

```
dtc = tree.DecisionTreeClassifier (random_state = 42)
parameters = [{"criterion":['gini','entropy'], 'max_depth': [5,10,15,30]}]
grid = GridSearchCV(dtc, parameters, verbose=5, n_jobs=-1)
grid.fit(x_train, y_train)
print("Best")
print(grid.best_params_)
print("Train score:", grid.score(x_train, y_train))
print("Validation score:", grid.score(x_validate,y_validate))
```

```
Fitting 5 folds for each of 8 candidates, totalling 40 fits
Best
{'criterion': 'gini', 'max_depth': 10}
Train score: 0.7790759870290509
Validation score: 0.7518427518427518
```

- Đầu ra có nghĩa là mô hình Cây quyết định đã được đào tạo và điểm số tham số tốt nhất là “criterion: gini” và “max\_depth: 10”.
- Điểm chính xác trên dữ liệu huấn luyện (train data) xấp xỉ 0.78, cho thấy mô hình đã dự đoán chính xác nhãn lớp cho 78% mẫu huấn luyện.
- Điểm chính xác trên dữ liệu kiểm định (validation data) là xấp xỉ 0.752, cho biết mô hình đã dự đoán chính xác nhãn lớp cho 75,2% tập kiểm định.

**Bước 2.** Khớp mô hình (Fit model) Cây quyết định (dtc) với tập dữ liệu huấn luyện ( $X_{train}$  và  $y_{train}$ ) nhằm tìm ra các tham số của mô hình sao cho nó có khả năng dự đoán tốt trên dữ liệu mới và in điểm đánh giá hiệu suất mô hình dựa trên dữ liệu huấn luyện và dữ liệu kiểm định cho các siêu tham số mặc định.

```
print("Default scores:")
dtc.fit(x_train, y_train)
print("Train score:", dtc.score(x_train, y_train))
print("Validation score:", dtc.score(x_validate, y_validate))
```

```
Default scores:
Train score: 0.9994818306421956
Validation score: 0.7203530060672918
```

- Điểm huấn luyện xấp xỉ 0,999 cho thấy mô hình dự đoán đúng 99,9% mẫu huấn luyện.
- Điểm kiểm chứng xấp xỉ 0,721 cho thấy mô hình đã dự đoán đúng 72,1% số mẫu kiểm chứng.

**Bước 3.** Tạo DataFrame từ kết quả tìm kiếm dạng lưới (grid search) và sắp xếp kết quả theo thứ tự tăng dần của điểm xác thực.

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

	mean_fit_time	std_fit_time	mean_score_time	std_score_time	param_criterion	param_max_depth	params	split0_test_score	split1_test_score	split2_test_score	split3_test_score	
1	0.047389	0.032870	0.010233	0.006402	gini	10	{'criterion': 'gini', 'max_depth': 10}	0.746532	0.743325	0.738728	0.738728	0
6	1.739481	0.053809	0.010623	0.000910	entropy	15	{'criterion': 'entropy', 'max_depth': 15}	0.741350	0.743920	0.747931	0.747931	0
2	1.467015	0.031051	0.010600	0.002456	gini	15	{'criterion': 'gini', 'max_depth': 15}	0.741935	0.740744	0.742917	0.742917	0
5	1.379526	0.053220	0.011006	0.002814	entropy	10	{'criterion': 'entropy', 'max_depth': 10}	0.730178	0.737819	0.761406	0.761406	0
7	1.821462	0.072725	0.008604	0.004456	entropy	30	{'criterion': 'entropy', 'max_depth': 30}	0.710430	0.709904	0.718763	0.718763	0
3	1.957101	0.027934	0.004362	0.002325	gini	30	{'criterion': 'gini', 'max_depth': 30}	0.708561	0.708483	0.710573	0.710573	0
0	0.570678	0.016639	0.006172	0.005033	gini	0	{'criterion': 'gini', 'max_depth': 0}	0.700318	0.693357	0.695529	0.695529	0

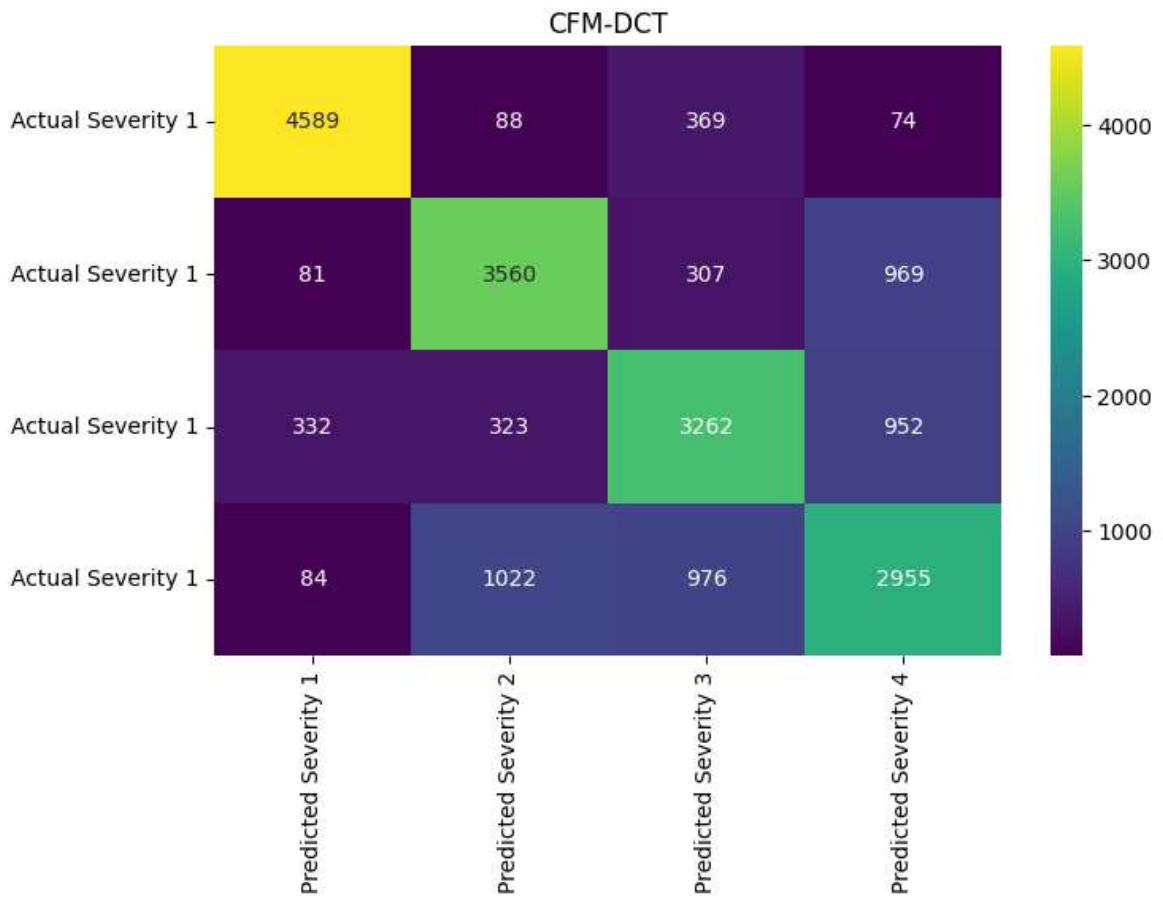
### Bước 4.

- Dự đoán các nhãn của bộ xác thực ( $X_{\text{validate}}$ ) bằng cách sử dụng mô hình cây quyết định (dtc) đã được đào tạo.
- Tính toán ma trận nhầm lẫn bằng cách so sánh các nhãn thực ( $y_{\text{validate}}$ ) với các nhãn dự đoán ( $y_{\text{pred}}$ ).
- Tạo ma trận nhầm lẫn để trực quan hóa hiệu suất của mô hình. Ma trận nhầm lẫn được lưu trữ trong Khung dữ liệu Pandas và được hiển thị bằng cách sử dụng bản đồ nhiệt từ thư viện Seaborn.

```
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
y_pred = dtc.predict(X_validate)
confmat = confusion_matrix(y_true=y_validate, y_pred=y_pred)

index = ['Actual Severity 1', 'Actual Severity 1', 'Actual Severity 1', 'Actual Severity 1']
columns = ['Predicted Severity 1', 'Predicted Severity 2', 'Predicted Severity 3', 'Predicted Severity 4']
conf_matrix = pd.DataFrame(data=confmat, columns=columns, index=index)
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix, annot = True, fmt='d', cmap='viridis')
plt.title("CFM-DCT")
plt.show()
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS

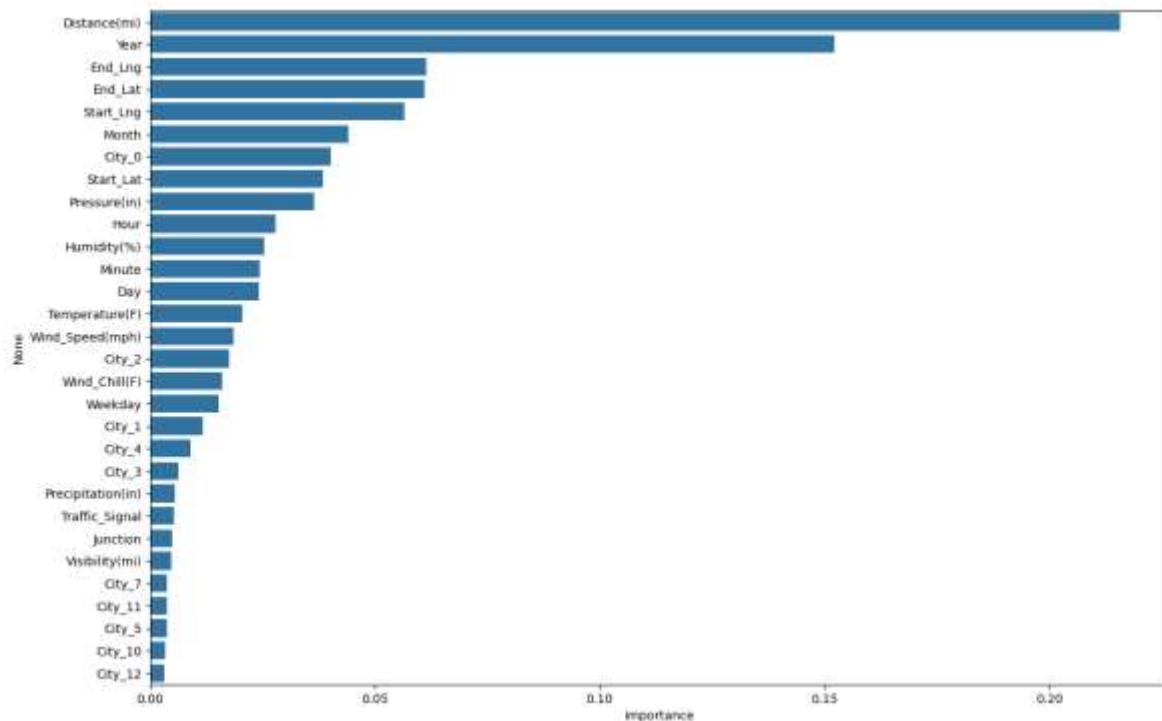


**Bước 5.** Tìm ra 30 tính năng quan trọng hàng đầu bằng cách tạo DataFrame có số hàng bằng số cột trong X\_train và một cột có tên là “quan trọng”, sử dụng tên thuộc tính trong X\_train làm chỉ mục.

```
importances = pd.DataFrame(numpy.zeros((X_train.shape[1],1)), columns = ['importance'], index=X_train.columns)
importances.iloc[:,0] = dtc.feature_importances_
importances = importances.sort_values(by='importance', ascending=False)[:30]

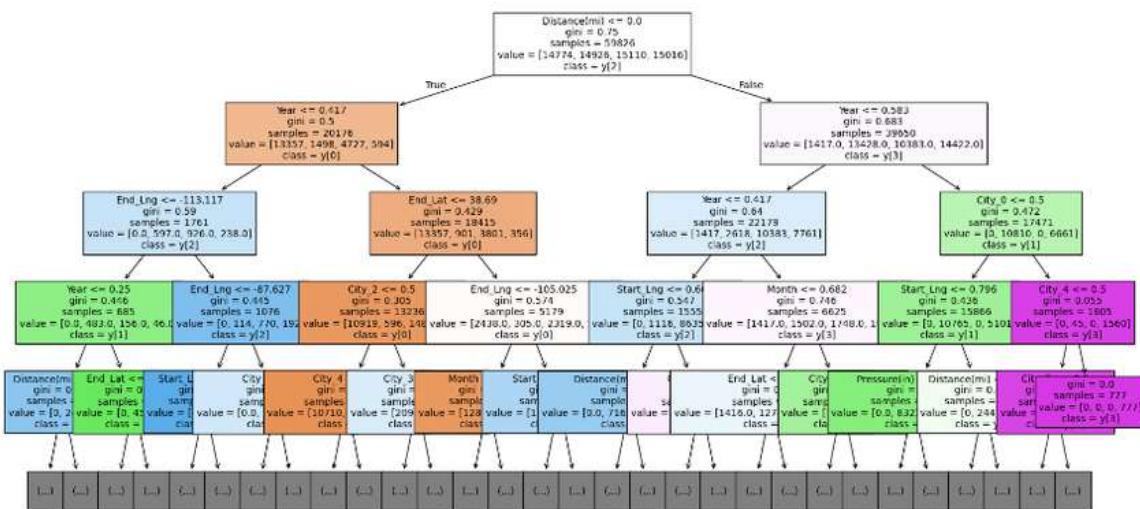
plt.figure(figsize=(15,10))
sns.barplot(x='importance',y=importances.index, data=importances)
plt.show()
```

## Đồ án xây dựng kho dữ liệu US ACCIDENTS



**Bước 6.** Tạo trực quan hóa mô hình Cây quyết định ‘dtc’ bằng cách sử dụng hàm plot\_tree() từ sklearn.tree, với max\_depth = 4.

```
from sklearn.tree import plot_tree
fig, ax = plt.subplots(figsize=(20, 10))
plot_tree(dtc, max_depth=4, fontsize=10, feature_names=X_train.columns.to_list(), class_names=True, filled=True)
plt.show()
```



### 4.2.3. Random Forest

Xây dựng và dự đoán với mô hình Decision Tree

Bước 1.

## **Đồ án xây dựng kho dữ liệu US ACCIDENTS**

- Thực hiện bộ phân loại Random Forest: Đoạn mã khởi tạo một bộ phân loại Random Forest (rfc) bằng cách sử dụng lớp Random Forest Classifier.
- Đôi tượng bộ phân loại Random Forest được khởi tạo với n\_jobs và random\_state.
- Các siêu tham số cần được điều chỉnh được chỉ định trong một từ điển được gọi là parameters.
- Một đôi tượng GridSearchCV được tạo ra và phù hợp với dữ liệu huấn luyện.
- Các siêu tham số tốt nhất được xuất ra cùng với các điểm số độ chính xác của mô hình trên dữ liệu huấn luyện và dữ liệu xác thực
- Kết quả đầu ra cho thấy rằng mô hình bộ phân loại Random Forest đã được huấn luyện và đánh giá bằng cách sử dụng GridSearchCV với 5-fold cross-validation trên 2 tổ hợp siêu tham số (n\_estimators, max\_depth).
- Điểm số độ chính xác trên dữ liệu huấn luyện là 0,999, cho thấy rằng mô hình đã dự đoán đúng nhãn lớp cho 99,9% các ví dụ trong tập huấn luyện.
- Điểm số độ chính xác trên dữ liệu xác thực là 0,827, cho thấy rằng mô hình đã dự đoán đúng nhãn lớp cho 82,7% các ví dụ trong tập xác thực.

**Bước 2.** Sử dụng mô hình bộ phân loại Random Forest (rfc) để phù hợp với tập dữ liệu huấn luyện ( $X_{train}$  và  $y_{train}$ ) và in ra các điểm số huấn luyện và xác thực với các siêu tham số mặc định.

- Điểm số huấn luyện là 0,999 cho thấy rằng mô hình đã dự đoán đúng 99.9% các mẫu huấn luyện.
- Điểm số xác thực là 0,824 cho thấy rằng mô hình đã dự đoán đúng 82.4% các mẫu xác thực.

**Bước 3.** Tạo một DataFrame Pandas chứa kết quả của cross-validation lưỡng tách kiểm và sắp xếp theo rank\_test\_score.

Kết quả DataFrame chứa một hàng cho mỗi tổ hợp siêu tham số và các cột cho các số liệu đánh giá khác nhau, bao gồm điểm số trung bình trên tập kiểm tra, độ lệch chuẩn của điểm số kiểm tra và các giá trị của siêu tham số.

**Bước 4.** Tạo ma trận nhầm lẫn để trực quan hóa hiệu suất của mô hình. Ma trận nhầm lẫn được lưu trữ trong một DataFrame Pandas và được hiển thị bằng cách sử dụng heatmap từ thư viện Seaborn.

**Bước 5.** Tạo biểu đồ cột để đại diện cho độ quan trọng của tính năng, được xác định bởi bộ phân loại Random Forest cho 30 tính năng hàng đầu. Điểm số độ quan trọng của một tính năng càng cao thì tính năng đó càng ảnh hưởng đến việc dự đoán với mô hình Random Forest.

**Bước 6.** Vẽ cây quyết định

Trong đó, ta sử dụng thuộc tính estimators\_ của mô hình Random Forest để lấy ra cây quyết định đầu tiên. Sau đó, ta sử dụng hàm plot\_tree để vẽ cây quyết định với các tham số tương tự như khi sử dụng với Decision Tree. Cuối cùng, ta sử dụng hàm show() của matplotlib để hiển thị cây quyết định.

## CHƯƠNG 5. TÀI LIỆU THAM KHẢO

- [1] GeeksforGeeks, "Difference between Star Schema and Snowflake Schema," 21 02 2023. [Online]. Available: <https://www.geeksforgeeks.org/difference-between-star-schema-and-snowflake-schema/>. [Accessed 10 04 2023].
- [2] V. K. Xindong Wu, "CART: Classification and Regression Trees," in The Top Ten Algorithms in Data Mining, CRC Press Taylor&Francis Group, 2009, pp. 179-183.
- [3] G. M. L. Education, "Classification: Accuracy," 19 07 2022. [Online]. Available: <https://developers.google.com/machine-learning/crash-course/classification/accuracy>. [Accessed 20 06 2023].
- [4] G. M. L. Education, "Classification: ROC Curve and AUC," 19 07 2022. [Online]. Available: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>. [Accessed 20 06 2023].
- [5] L. Breiman, "SpringerLink," 10 2021. [Online]. Available: <https://link.springer.com/article/10.1023/a:1010933404324>. [Accessed 15 06 2023].

***Đồ án xây dựng kho dữ liệu US ACCIDENTS***